The possibilities are endless...

FPS Creator X10 - Version 1.0

USER GUIDE

© 2005-2007 The Game Creators Ltd. All Rights Reserved.

Contents

1 Introduction 6

2 System Requirements 7

3 Installation 8
3.1 Install process 8
3.2 Auto update system 9

4 Getting Started 10
4.1 The Game Level Editor 10
4.2 Editor Key List 12
4.3 Making Your First FPS Game 13
4.4 Making Your First Arena Game 20

5 The Media Libraries 23
5.1 Prefabs 23
5.2 Segments 24
5.2.1 Segment Overlays 26
5.3 Entities 29
5.3.1 Entity Properties 31
5.3.2 Entity AI 32
5.3.3 Weapon Entities 33
5.3.4 Ammo Entities 35
5.3.5 Enemy Entities 36
5.3.6 General Entities 38
5.3.7 Description of the general FPI scripts 39
5.3.8 How Doors and Switches Work 41
5.3.9 Entity Explosions and Debris 42
5.3.10 The FACTOR Setting 42
5.3.11 Dark AI Scripts 42
5.4 Markers 43
5.4.1 Player Start Position Marker 44
5.4.2 Player Checkpoint Marker 44
5.4.3 Light Markers 45
5.4.4 Zone Markers 47
5.4.5 Story Marker 48
5.4.6 Stretching a marker over layers 48

6 More Map Editing Information 49
6.1 Cursor Editing States 49
6.2 Editor Indicators 50
6.3 3D Clipping Modes 50
6.4 Special Media 51
6.4.1 Gantries 51
6.4.2 Lifts 52
6.4.3 Teleports 52
6.5 Waypoints 53
6.5.1 Creating a New Waypoint 53
6.5.2 Linking Waypoints 54
6.5.3 Waypoints across Layers 54
6.5.4 Waypoint Command List 55
6.5.5 Waypoints for characters moving between layers 55

7 Testing And Previewing Levels 56
7.1 Test Level 56
7.2 Effects Menu 57
7.3 Volume Menu 59
7.4 Performance Monitor 59
7.5 Test Game Preferences 60
7.6 Quick Level Preview 61

8 Building A Stand Alone Game 62
8.1 Game Project Settings **63**
8.2 Level Settings **64**
8.3 Textures **68**
8.4 Arena Executables **70**
8.5 Build Settings **71**

9 Physics 72
9.1 How Physics Works **72**
9.2 Creating physics based situations **72**
9.3 Tricks and tips when setting up physics **73**

10 Multiplayer Death Match Games 74
10.1 How Multiplayer Works **74**
10.2 Creating a Multiplayer Game **74**
10.3 Differences between Arena and Single Player **75**

11 Guidelines 76

12 GUI Layout 78
12.1 The Drop Down Menus **78**
12.2 Toolbars **81**

13 How To Guides for Advanced Users 83
13.1 Game Resolution **83**
13.2 Making New FPI Logic **83**
13.3 Making New Segments and Entities **90**
13.4 Making Your Own Weapons **90**
13.5 Customizing The In-Game Screens **92**
13.6 Description of the Files Folder **93**
13.7 Overview of Files and Formats **94**

14 Expand your model pack library 95

15 Other Game Making Tools 106

Credits

FPS Creator X10 was thought up and lovingly developed over a six year period by the following individuals:

Lead Programmer	Lee Bamber
Programming	Mike Johnson, Paul Johnston
Project Manager	Richard Vanner
Lead Artist	Simon Benge
Artists	Mark Blosser, Jonathan Fletcher, Michael Palm
Animation	Simon Benge, Chris Bamber
2D Art	Peter Jovanovic
Music	Chris Blundell, Willem Cramer
Web Site	Richard Davey
Voice Over	Lee Bamber Sarah Nielsen courtesy of Chris Blundell Shockwave-sound.com
Weapon Sound Effects	Rick Harrison
Environmental Sounds	Mark Sheeky
Manual	Richard Vanner, Lee Bamber
Technical Support	NVIDIA Corp (especially Roy Taylor, Phil Scott, Phil Wright, Bill Rehbock and Leslie Pirritano)
Testers	Kristin Bjørli, Terry Schwarz, Kyle Waite, Martin Eaton, David Hutchison, Paul Didcote, Jonathan Smit, Michael McMullen, Andrew Vanbeck, Dave Milton, David Johnson, David Tattersall, Frederic Cordier, Ian Mold, Jake Jensen, Jason Dean, Jess Telford, Robert Knight, Ron Erickson, Malcolm Bamber, Alec Moody, David Smith, David Tattersall, Chris Blundell, Christian Mueller, Peter Coleman, Chris Breaux, Evan Floyd, Stephen Jolly, Eric Tomassetti, Brentt Moore, Timothy Ruswick, Oleg Sergeyev and Alex Wekell.

FPS Creator was developed using an internal development version of Dark Basic Professional X10. For more information on Dark Basic Professional and our full product range, please visit:

http://www.thegamecreators.com/

Contact Information

Support

Visit the dedicated web site for the latest FPS Creator X10 news, add-ons, advice and much more:

www.fpscreatorx10.com

Join the community!

Find out what other users are doing with FPS Creator X10 by registering and participating within the FPS Creator X10 online forum:

http://forum.thegamecreators.com/

FPS Creator X10 uses the ODE Physics Engine

ODE's BSD license (LICENSE-BSD.TXT)

Open Dynamics Engine
Copyright (c) 2001-2004, Russell L. Smith.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the names of ODE's copyright owner nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For more information, visit:

http://www.ode.org/

1. Introduction

The idea for FPS Creator X10 was first conceived back in 2001 – it has taken over six years to bring this project to completion. Developing a tool that makes games is a fraught business; we have to cater for all kinds of complex issues in order to ensure fast game play and end-user enjoyment. The result of all this hard work is this second generation of FPS Creator specifically for Direct X10 PCs.

"Five years in the making, FPS Creator X10 has taken the next step in gaming technology to offer you even more power. The new DirectX 10 engine will make your games faster and unleash stunning visuals onto the player. The new DarkAI component breathes life into your game character, and provides allies who will fight alongside you. Multi-core features accelerate the building and real-time execution of your games, including the ability to capture videos of your game as you play. Be one of the first people to make DirectX 10 games for Windows Vista, and share your creations with our active FPS Creator X10 community!"

Lee Bamber
Lead Programmer and founder of The Game Creators Ltd

"The awesome power of DirectX 10 is now in your hands – this game making tool will allow you to harness the brute force power of your DirectX 10 graphics card. In the past only expert developers have had the inside knowledge on how to get the best out of GPUs. Now you can gain direct access to effects like bloom, reflective water, DirectX 10 instancing of objects, texture arrays and many more cutting edge technologies. The TGC team look forward to seeing what you create – you hold true power in your hands, be careful how you use it!"

Rick Vanner
Financial Director, The Game Creators Ltd

2. System Requirements

IMPORTANT: FPS Creator X10 requires DirectX version 10.0 or higher to be installed prior to use. If you do not have the latest version you can download it for free here:

www.microsoft.com/directx

Recommended
Window Vista (any version)
Pentium 4, 2.66 Ghz Dual Core (or equivalent)
2GB RAM
DVD ROM Drive
2.0 GB of Hard Drive Space (+space for created games)
DirectX 10.0
DirectX 10.0 compatible Sound Card
DirectX 10.0 compatible Mouse and Keyboard
DirectX 10.0 compatible Video Card (3D Hardware Accelerated) with at least 768MB of memory

Minimum Requirements
Window Vista (any version)
Pentium 4, 2Ghz (or equivalent)
1GB RAM
DVD ROM Drive
2.0 GB of Hard Drive Space (+space for created games)
DirectX 10.0
DirectX 10.0 compatible Sound Card
DirectX 10.0 compatible Mouse and Keyboard
DirectX 10.0 compatible Video Card (3D Hardware Accelerated) with at least 256MB of memory

3. Installation

3.1 Install process

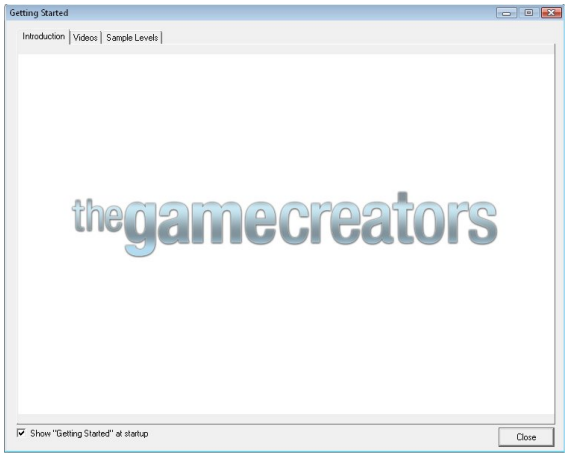
Installing FPS Creator X10 is very straight forward:

- First insert the DVD into the DVD drive.
- The install process with begin if Windows is still setup to auto run the DVD. If it's not, then run the file setup.exe
- You will be asked to enter your name and serial number. The number will be printed on a label, found on the inside of the software DVD case.

InstallShield Wizard
User Details
Please enter your Name and Serial Number. Use your serial number from FPS Creator. If you enter an invalid serial code, the name field will be cleared and you must enter a correct serial number.
Name:
Serial:
InstallShield
< Back
Next >
Cancel

When the installation is complete you can find FPS Creator X10 on the Start/All Programs menu in the group labelled "The Game Creators":

After you run the main application you will be presented with the "Getting Started" screen:



You can choose to watch the various videos, select a sample game level or close the box and start editing from scratch.

3.2 Auto Update System

The version of FPS Creator X10 on the DVD might have been superseded with a more up to date version. FPSC X10 has a built in version. You can access the "Check for Updates..." feature from the Help Menu:

Automatic Update
The auto update feature will contact the TGC server and detect the latest available version of FPS Creator X10. If a new version is found, you can click the download button to automatically install it to your computer.
If you have a firewall active, you must allow FPS Creator X10 to access the internet or the update will fail.
Check at start-up
Current Version: n/a
Latest Version: 1.08
Checking for updates...
X10
thegamecreators
Check For Updates
Download Update
Close

The automatic update dialog will allow the software to update itself as new versions become available. When the software runs for the first time, it will ask you whether you would like to check for the latest version via the automatic update dialog.

We appreciate you may not want the software to contact the Internet right away, or have security concerns, and the software will only communicate with the FPSC server if you want it to. When you enter the automatic update dialog, you will notice a tick-box which you can use to deactivate the check for new versions when the software subsequently starts. When the software does start with this tick-box active, it will automatically communicate with the server and compare the latest version with the one you have installed. If you are running the latest version, the software will do nothing more in this regard. If a new version is detected, it will open up the automatic update dialog allowing you to download and install the update. A status box is provided inside the automatic update dialog to inform you exactly what is happening at every stage, along with a progress bar to indicate the progress of your download.

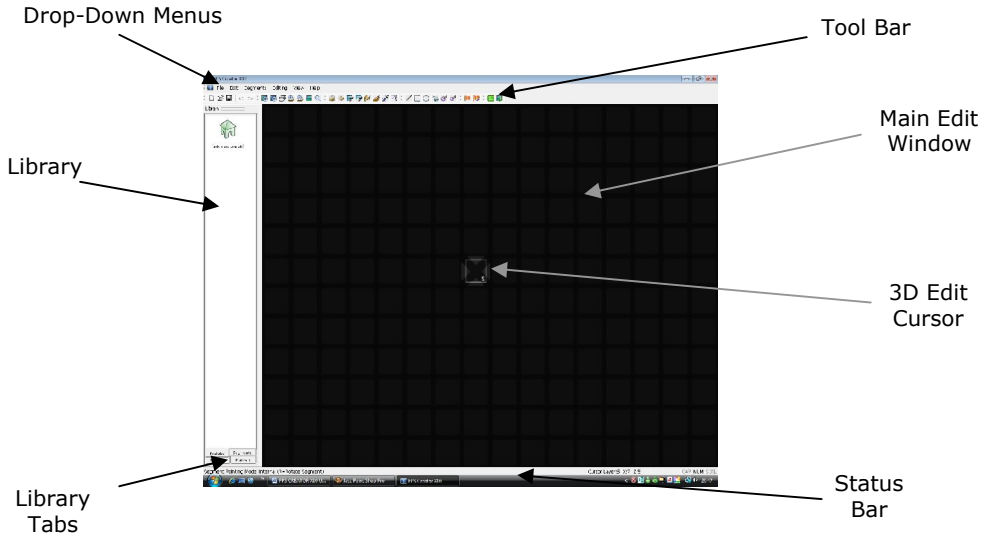
If you wish to know what changes are included in any new update, you can do this easily by exiting the automatic update dialog and clicking the web link from the Help menu, from where you can navigate to the Upgrade section of the FPSC X10 website. If you have a firewall system running on your computer, you will be asked whether you want to unblock the FPSC software and allow it to communicate with the outside world. You must select unblock otherwise the software has no way of checking for versions or downloading the update

4. Getting Started

4.1 The Game Level Editor

We know that the first thing you want to do is jump right in and create a game and even though we have made the whole process as easy as possible, there are still things you need to be aware of before you start. The best way to move forward is to work your way through these simple tutorials. They will introduce you to many of the systems used within FPS Creator X10. We have also provided video tutorials to get you up and running.

When you first run FPS Creator X10 you will be presented with this screen:



Tool bar
Icons in this bar provide quick access to all of the important features of the software.

The drop-down menus
Access further options by selecting the various drop-down menu items.

Library
This window displays thumb nails of the building blocks used when making your games. Here you will select from the various *segments* and *prefabs* that allow you to construct game level geometry and from the *entities* that you've selected to appear in your game.

Library Tabs
The Library is split up into four sections: Prefabs, Segments, Entities and Markers. Click a tab to select which area of the Library you are currently viewing.

Main edit window
Displayed within this area is a top down view of the game level you are currently editing. To make the process of level editing as easy as possible FPS Creator X10 uses a grid based system for building 3D worlds. Each cell of the grid is a 100x100x100 game world cube. You will see later that you can choose special segment building blocks that can be joined together to make levels. If you are in Arena mode, this grid will be displayed with a tint of red colour to help you easily recognise the editing mode you are in.

The 3D edit cursor
When you paint into the game level the 3D cursor shows you where this painting will be applied. PCs and the new DX10 GPUs are very powerful. With this extra power we are able to render a lot of the game's scene from a bird's view. If you find your PC is struggling to do this (unlikely), you can decrease how much of the scene is rendered by pressing the ` key.

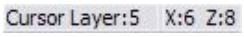
If you do decrease the render area you will notice a black circle surrounding the cursor, this is called the render shroud. The shroud gives you a rendered view of the immediate area you are working on and the rest of the map is quickly drawn with a blue print image.

Status Bar
At the base of the edit screen is the status bar. This displays the current edit mode you are using, your cursor's edit location within the level and the status of the scroll, caps and num lock keyboard states.

The range of game level size is as follows:

Description	Size	Axis	Cell numbering	Keys for moving
Height	20	Y	0-19	"+" and "-"
Width	40	X	0-39	Left and Right Arrow
Length	40	Z	0-39	Up and Down Arrow

The status bar shows your 3D cursor current location:



In this example the cursor is in Layer 5, 6 cells from the left edge of the grid and 8 cells down from the top of the grid. To understand what a layer is, imagine an apartment block, the ground floor would be layer 0. Floor 1 would be layer 1, floor 2 layer 2 and so on. The editor gives you a bird's eye view of a layer and presents the current layer (floor) that you are working on, hiding the other layers by default.

IMPORTANT NOTE: Be aware that the default editing layer is layer 5. We have done this because you may find that you want to expand your level down. If you had started editing at layer 0 then you would have no room to manoeuvre. You can of course start at layer 0, you just keep pressing the - key until you reach the desired layer and start editing from there.
The first time FPS Creator X10 begins it is set ready for you to make a new single player based game. It's important for you to realise that you must choose early on which type of game you are to make, a single player FPS or a Multiplayer death match arena game. It is still possible to switch between these game types, it will just help you to be set in the correct mode from the start. See the section on Multiplayer gaming for more details on making Arena type games.

4.2 Editor Key List

MAP MOVEMENT/Display	
+	Move up a layer
-	Move down a layer
Shift+Mouse wheel	Move up and down layers
Z	Zoom view
<	Zoom into map
>	Zoom out from map
Mouse wheel	Zoom in and out of map
#	Increase size of render shroud
`	Decrease size of render shroud
Ctrl+Mouse wheel	Increase/decrease render shroud

GENERAL KEYS	
Left Mouse	Paint
Right Mouse	Delete
R	Rotate selection by 90 degrees
Ctrl+R	Rotate selection by 10 degrees
Shift+R	Rotate selection by 1 degree
Ctrl+Z	Undo
TAB	Clip 3D view (3 modes)
Arrow keys	Move 3D cursor
Shift+Arrows	Move 10 cursor positions
Ctrl+Arrows	Move to end of map grid
F1	Quick key help

SEGMENT EDITING	
P	Segment editing
K	Pick segment from map
A	Paint walls & floor in interior mode
X	Paint walls & floor in exterior mode
M	Paint walls only
F	Paint floor only
E	Clipboard Edit mode
W	Way point editing mode
Space Bar	View waypoints on/off toggle

ENTITY EDITING	
N	Entity Editing
B	Grid mode (3 modes)
Y	Set Entity to Static (red) or Dynamic (green)
Left Mouse	Select and Paint entity
Right Mouse	Edit properties
Enter	Moves the entity on top of the highest entity
Shift+Enter	Moves the entity on top of any entities below
Page Up/Down	Move entity up and down the Y axis
Arrow keys	Move entity relative to the camera position

CLIPBOARD EDITING	
Ctrl+P	Paste clipboard
Ctrl+C	Copy area
Left Mouse	Drag selected area

ENTITY PROPERTIES	
Right Mouse	Drag to rotate view
Left Mouse	Exit edit properties

MARKERS	
[	Decrease light range
]	Increase light range

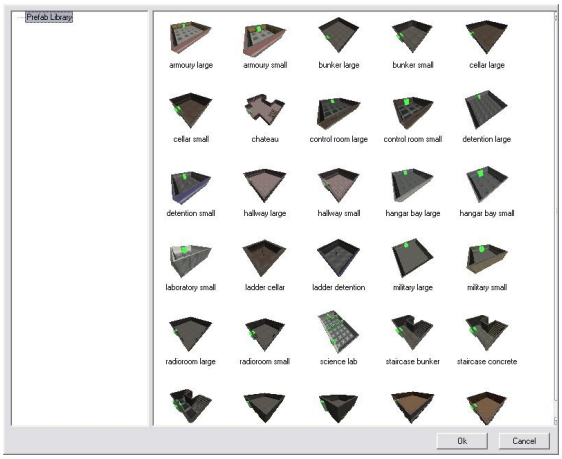
4.3 Making our First FPS Game

Let's create our first game by following the steps below. Remember that if you make a mistake while editing you can use the Undo icon / menu item (or press Control+Z) and this will reverse the last action, avoiding you having to start all over again.

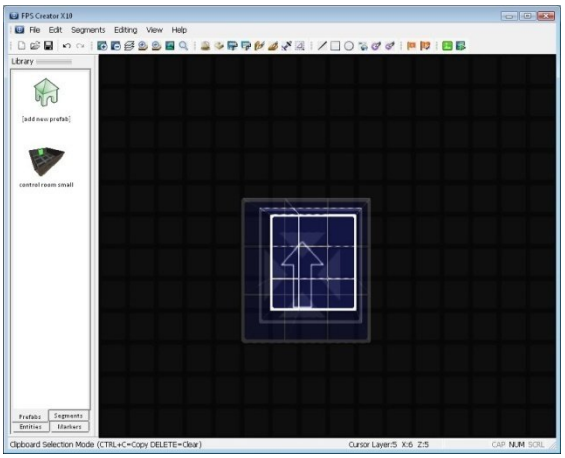
All of the maps for these tutorials were installed along with FPS Creator X10. So if you get confused by a tutorial you can simply load in the appropriate map which will be referenced like this:

> 🗁 "FPSC X10 Files/Mapbank/Tutorials/Tutorial_01.fpm"

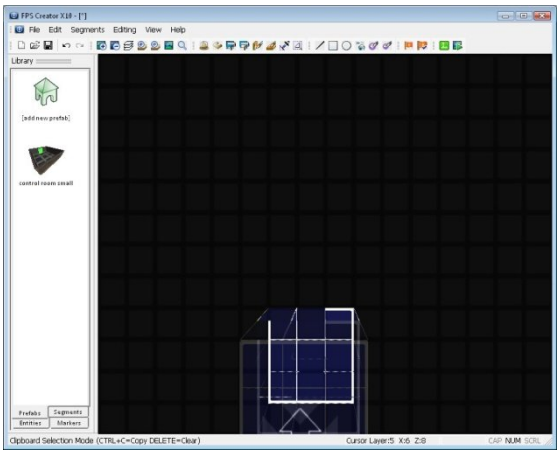
1. To get started we need a building for our game. In the Library window, click on the "add new prefab" icon , select "control room small" and click OK.



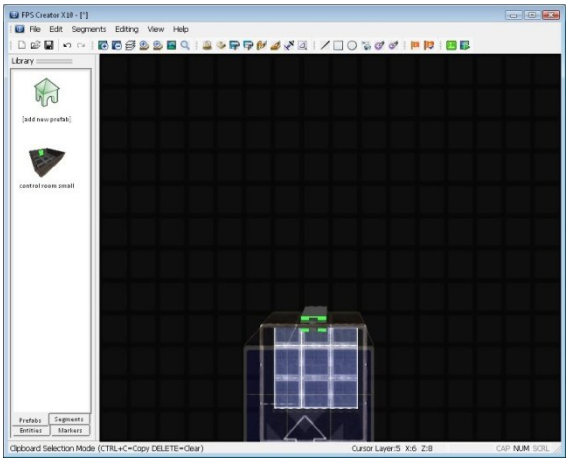
2. The Prefab is now attached to the mouse. Move your mouse around and you will see a blue print image of the prefab trailing it. The prefab is made up of segments and entities and this one is more than a layer high.



3. Position the prefab so that it's at the base of the screen like this:



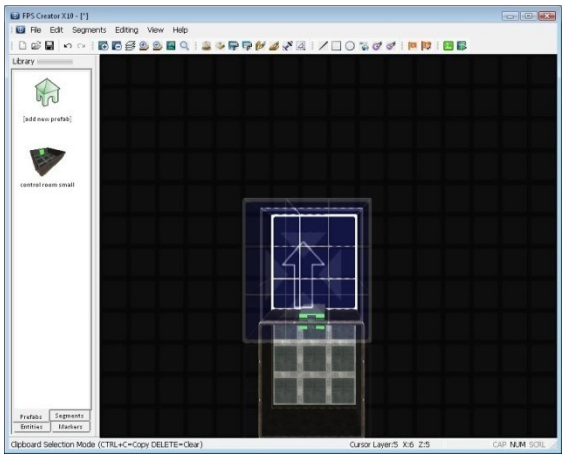
4. Now click the left mouse button to place the prefab into the level:



🗁 "FPSC X10 Files/Mapbank/Tutorials/Tutorial_01.fpm"

One thing to notice about the room you have just created is the colour of the doorway. In the editor it shows up as green. This is because at this point the door has not been cut into the wall of the room. The cutting process happens when the game is tested or built.

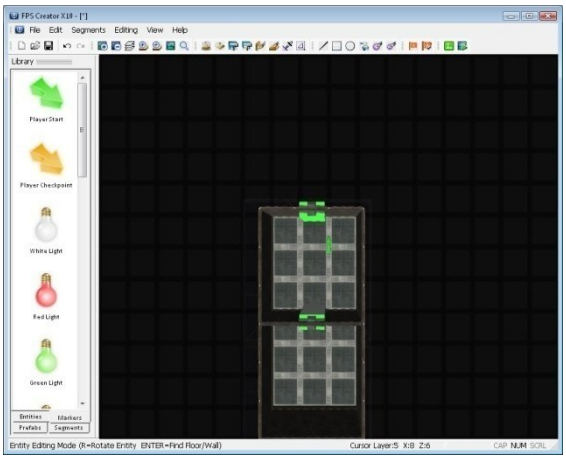
5. The prefab is still available for placing again. So now position the mouse above the room you have just pasted. Make sure the two rooms are perfectly aligned and then press left mouse to paste another instance of it:



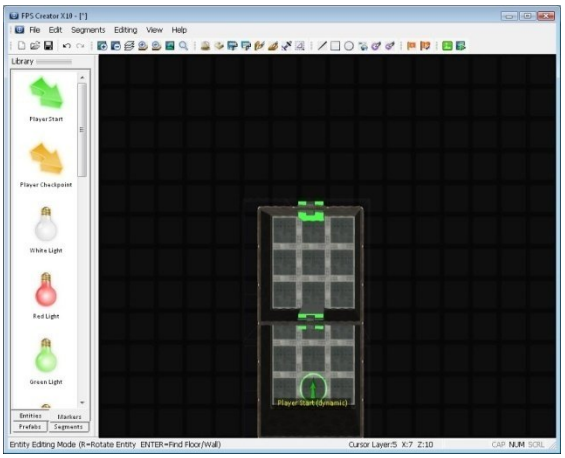
🗁 "FPSC X10 Files/Mapbank/Tutorials/Tutorial_02.fpm"

6. We now have a game level consisting of two joined rooms. Let's define where the player will start when the game is first run.

7. Click on the Markers tab at the base of the Library window and then click on the green arrow marker labelled "Player Start":

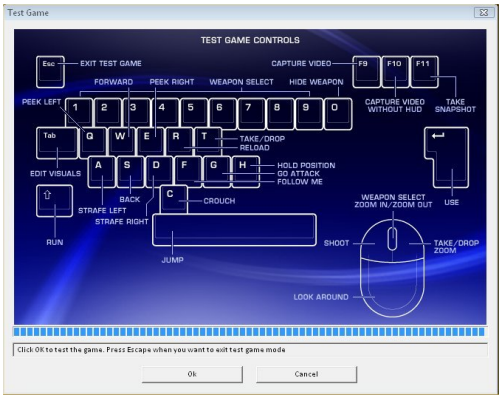


8. Position the player start position into the first room at the base of the screen. You can rotate the angle of the start position by pressing the "R" key. Left click when you are happy with its position:



🗁 "FPSC X10 Files/Mapbank/Tutorials/Tutorial_03.fpm"

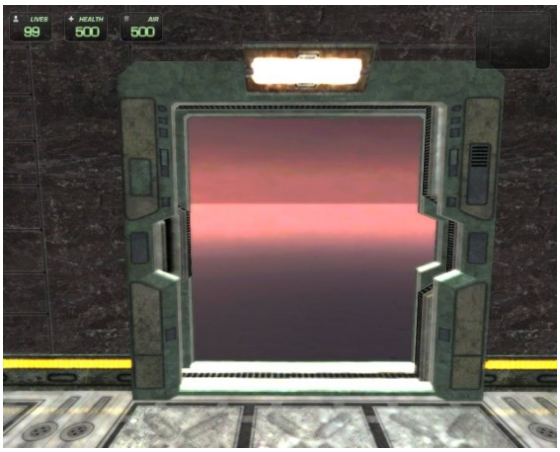
9. OK, let's test out our level. From the tool bar, click on the Test Game icon: This will begin the process of building your game from the selections you made. Once complete you'll see a window similar to this:



10. Click OK and the game will run. Depending on where you placed the player start marker (and if you rotated it or not) you should see something similar to this:

11. Use the W/A/S/D keys to move forward, back, left, and right. Use the mouse to look around. The prefab we used had a door, move towards the door and it will open automatically:

12. Walk into the next room and you can even walk up to the other door and see outside of the two rooms you have created:



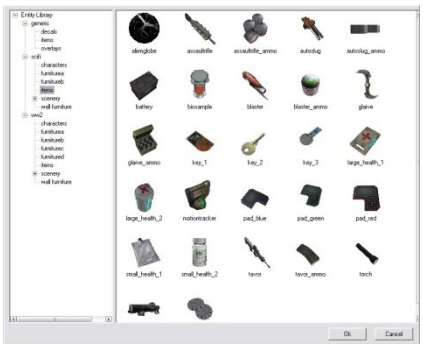
Of course when you design your own levels you would be careful to ensure that a player wouldn't walk out of a room into space – unless you wanted that to happen of course!

Now it's time to add some action!

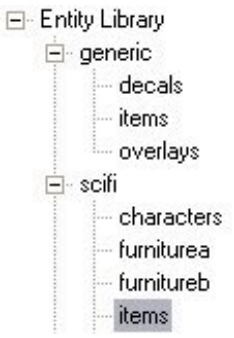
13. Click on the Entities tab at the base of the Library window.

14. Now click on the "add new Entity" icon:

15. The browser window will display covering up the map editing area. This window should be familiar to anyone who uses Windows – on the left you have the categories available and on the right are thumb-nails of the items within the current category. By expanding the categories on the left, navigate until you can see the contents of "Entity Libray/Sci-Fi/Items":



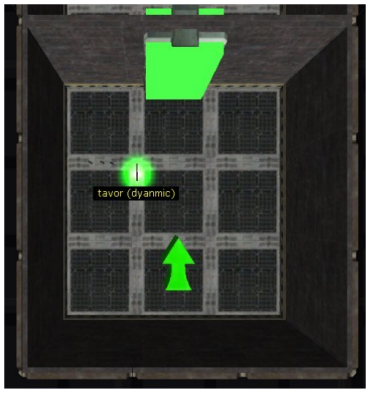
The browser window displaying the items from scifi genre



The navigation tree for finding Entity Library/scifi/items

16. Select the "Tavor" weapon item and click OK (or just double click the thumb-nail).

17. Place this weapon within the first room as shown here:



18. Time to add some ammo for our weapon. Again click on the "add new Entity" icon and this time select "Tavor_ammo" from the same category.

19. Scatter a few ammo pick-ups near the gun, three would be a good amount.

    🗁 "FPSC X10 Files/Mapbank/Tutorials/Tutorial_04.fpm"

20. Now it's time to add an enemy! Click on "add new Entity" and browse to point to "Entity Libray/Sci-Fi/Characters". From the list choose "Raknos (Shotgun)".

21. You are now adding an enemy into your map. Place the Shot gun wielding alien into the top room (and use the R key to rotate him so that he's facing south towards the door the player will walk through).

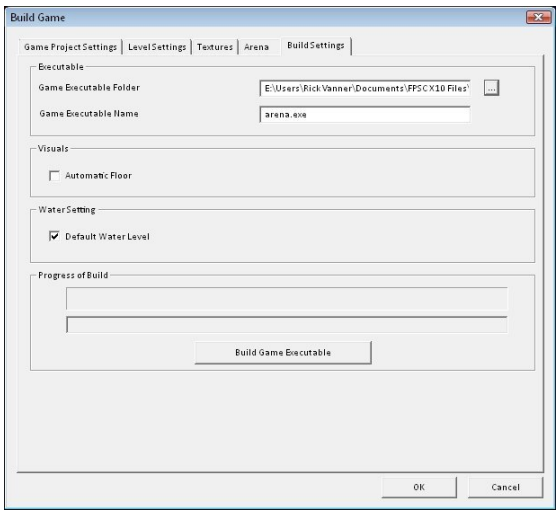
    🗁 "FPSC X10 Files/Mapbank/Tutorials/Tutorial_05.fpm"

22. Like the door, the enemy character is already programmed with an intelligent AI script. He has been instructed to kill you on sight! Let's see if your gaming skills are up to scratch. Click on the "Test Game" icon to rebuild the game and then play. Use the "R" key to reload your gun..

23. Pretty cool huh? In just a few steps you have built a tiny game level, selected a weapon and a bad guy to attack. All without a line of code in sight!

25. Now for something even cooler! First save your game using the File menu and call it mygame.fpm Now choose the option "Build Game" from the File menu. When the dialogue box appears, click on the tab "Level Settings". Under the text "List of Levels" there should be a level file listed, click on the name of this file and up will pop a file selector. Now select your game file "mygame.fpm". This tells FPS Creator X10 that you want to make a one level game using the map you just designed.

26. Now click on the "Build Settings" tab and then click on "Build Game Executable". FPS Creator X10 will now create a stand alone version of your game. This means you can pass your game onto your friends and let them play what you have made.



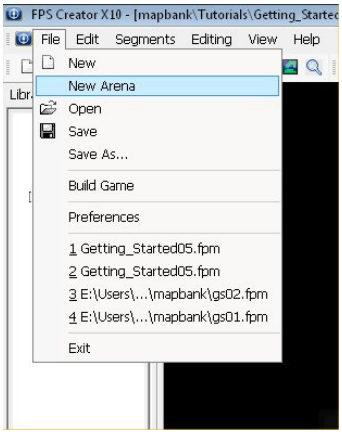
27. After the build process is complete. Use Windows Explorer and locate the directory where the game was created (the default location is C:\Documents\FPSC X10 Files\X10 executables\simple\simple.exe).
    Double click on the exe file and the game will load in and run with its own game menu structure.

4.4 Making your first Arena game

An arena game differs from a regular single player game in a number of ways. The most obvious is that it is designed to be played against other players found online. As such there is no story elements to the game, and is designed around quick game play, re-spawning weapons and ammo, and basic death match objectives. Follow these simple steps to create your first ever multiplayer arena game (make sure you have a friend or two to play against)!

Make sure you allow the game to send data to and from your PC via any firewall that may be setup on your system, otherwise the game packets will be prevented from being sent. If you have a router you need to open port number 2303. FPSCX10 also requires that port to use TCP & UDP protocols.

1. Select NEW ARENA from the FILE MENU

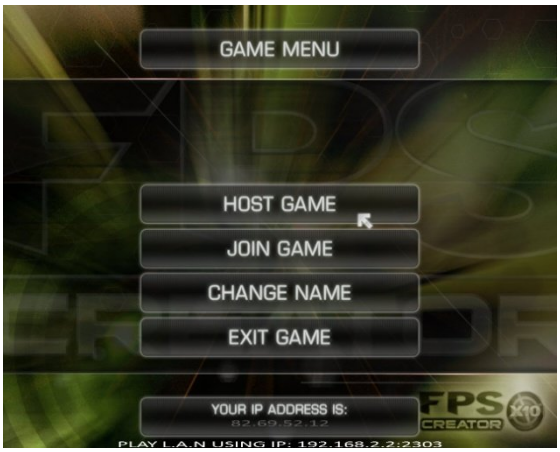


2. Select ADD NEW PREFAB and select any room.
3. Paint a room of about 8x8 segments as your arena level.
4. Select the ENTITIES tab and select ADD NEW ENTITY.
5. Choose any character entity and click OK.
6. Paint the character to the level.
7. Return to the ADD NEW ENTITY and select a second character.
8. Paint this second character into the level.
9. Select ADD NEW ENTITY and select a weapon.
10. Paint the weapon twice into the level.
11. Save the level, calling it myarena game.fpm
12. Click the BUILD GAME from the FILE MENU.
13. Click the LEVEL SETTING Tab and change the first level to use the file myarena game.fpm
14. Click the ARENA tab and edit the first to X frags with a value of 5.
15. Click the BUILD tab and ensure the build game executable name is set to myarena game.exe
16. Click on the BUILD GAME EXECUTABLE button.
17. View the X10 Executables folder using Windows Explorer.
18. Look for the "myarena game" folder and run the executable inside to play.

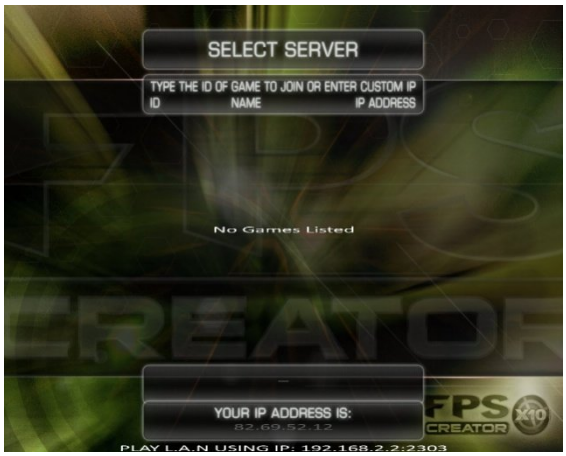
How to Play a Multiplayer Game

1. Run the arena executable to start the game.
2. You will be required to enter your name, which will be used in the game.

3. The player with the fastest machine should click the HOST button.
4. The players who wish to join in must click the JOIN button.



5. The joiners must also select the game they wish to play from the given list. Simply type the number of the game you wish to join and press Enter.



6. A character selection screen appears next. The host chooses first and then all other players must choose their character, on a first in basis.

7. When the game is completely loaded, the game begins immediately.

8. The winner is determined by whoever completes the objectives setup when the game was created (from the build game – arena settings).

You can press Escape at any time to leave a MP game. If the Host player escapes then all the Joiners will be sent back to their game's main menu.

This getting started section has introduced you to some of the key areas of the editor. There's plenty more to discover, and we'll continue in the next chapter by taking a more in-depth look at the various media libraries.

If you have trouble with connecting we recommend you visit our forums where you will be able to get help and advice on multiplayer gaming.

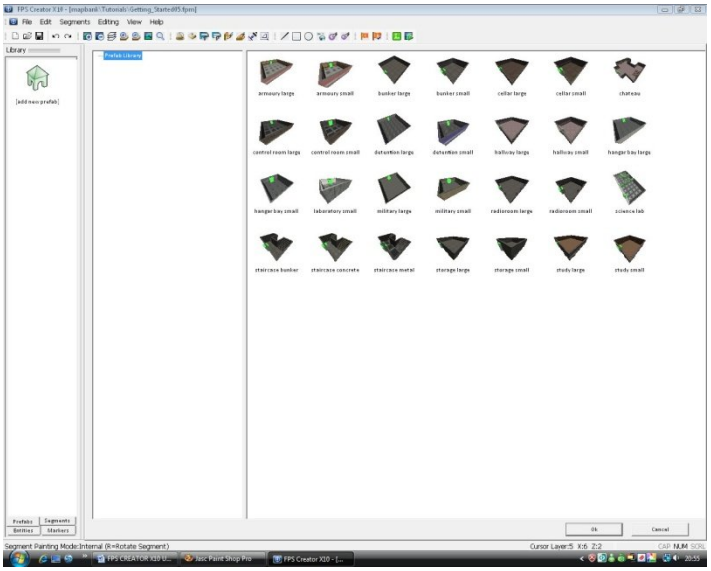
5. The Media Libraries

The media libraries are the hub of your game building experience. They contain four different types of game item: Prefabs, Segments, Entities and Markers.

5.1 Prefabs

Prefabs are constructed from the smaller segment building blocks. The idea behind using them is to save having to reconstruct similar geometry for your games time and time again. For example if you have created a really good room layout from the various segments provided then you can save this as a Prefab (see section 5.1.1) In this version of FPS Creator X10 we have constructed some prefabs to get you started. We also hope that many users will share their prefabs via the web site.

When you click on the "add new prefab" icon you will see this browser:



Once you have chosen a prefab it will be shown in blue print form under your mouse. You can now paint this prefab into the level.

Key things to note when you are pasting a prefab:

- The prefab's bottom layer, top left segment is drawn to the grid cell your 3d cursor is pointing to.
- Most prefabs are taller than segments and can be taller than a single layer, so be careful not to paint over areas higher up in your level.
- You can press "R" to rotate your prefab 90 degrees.
- The Right mouse button will delete all the cells covered by the prefab
- You can remove the prefab from your mouse pointer by pressing the "Delete" key.
- You can revert back to Segment painting by pressing "P" or to Entity editing by pressing "N".
- If you make a mistake, you can always press Control+Z to undo a paste.
- Prefabs cannot include standalone entities. They can include doors and windows (which indirectly contain some fixed entities).

5.2 Segments

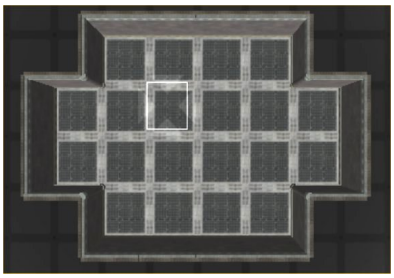
Segments are the raw materials you use to build your game levels in FPS Creator X10. They come in different shapes and sizes and you need to understand how they work together to build levels.

Painting with segments
Let's start with the basic room segment. A room segment comprises of information describing how it will be used when drawing walls (both internal and external), the floor and the ceiling.

If you choose the segment "Control Room Full" from the media library (see: Segment Library/Scifi/Rooms). Then draw a small 4x4 room with the segment, it will look like this:

🗁 "FPSC X10 Files/Mapbank/Tutorials/Segments01.fpm"

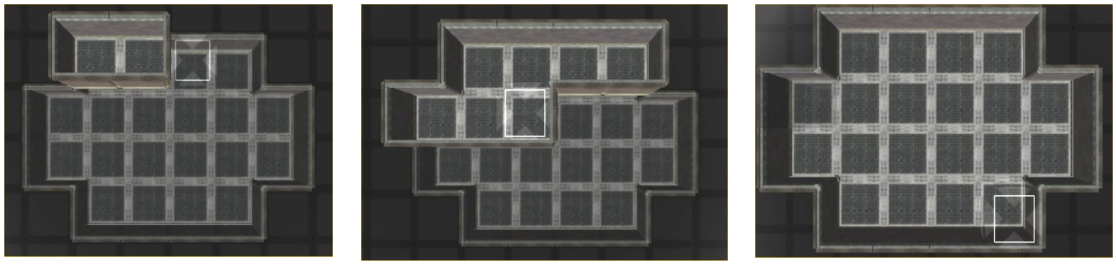
You will notice that the editor automatically constructs and joins the walls for you. This makes editing rooms very easy indeed. The segment also creates a floor for you. Let's add two extra cells to the left and right of the room:



🗁 "FPSC X10 Files/Mapbank/Tutorials/Segments02.fpm"

Again, the walls are calculated automatically. But what about a ceiling? By default, segments do not draw ceilings. This is because the ceiling belongs to the layer above the one we are editing. In fact, the segment has drawn a ceiling – it's made one for the room below the level we are editing (i.e. the floor).

Press the "+" key and move up to the next layer. Now place down the same segment over each cell you see below, like this:



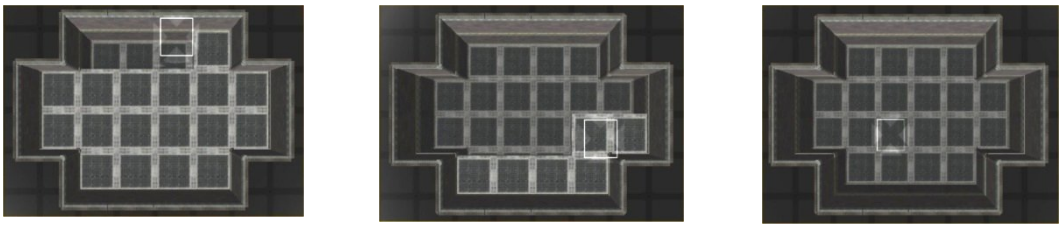
Sequence showing the segment being painted directly over the layer below

🗁 "FPSC X10 Files/Mapbank/Tutorials/Segments03.fpm"

You should have covered all of the cells so that you cannot see the room below. This has created a ceiling for the room on the first layer by creating another room above it – however you can also just draw on a ceiling as we cover in the next section.

Painting just floors and ceilings

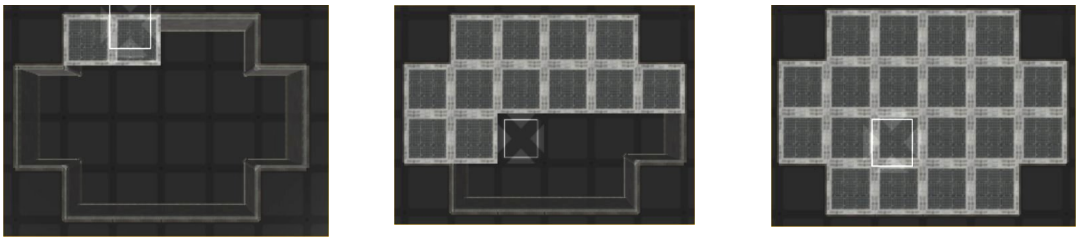
Press the "F" key on the keyboard. This sets you into Floor paint mode. Now RIGHT click on each floor piece to delete just the floor of each segment at this layer:



Sequence showing the floor of this layer being removed

🗁 "FPSC X10 Files/Mapbank/Tutorials/Segments04.fpm"

Notice how you can see the higher walls that now span over two layers.
Press the "+" key to move up another layer. Now LEFT click on each cell to add a ceiling above the two layers below, like this:

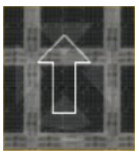


Sequence showing a floor & ceiling being added to this layer

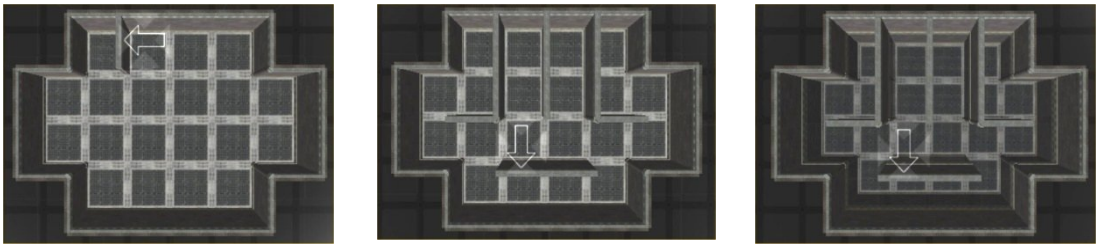
🗁 "FPSC X10 Files/Mapbank/Tutorials/Segments05.fpm"

Painting just walls

We will now change draw mode. Press the "-" key twice to return back to layer 5. Press the "M" key. You are now in Wall paint mode:



Notice that the cursor shows an arrow pointer. This indicates where the wall will be placed when you next click the left mouse button. You can rotate the arrow 90 degrees by pressing the "R" key. See if you can create the wall layout as shown below. When you have designed layer 5, press "+" to move up to layer 6 and repeat your wall painting:



Sequence showing walls being drawn. First in layer 5, then in layer 6

🗁 "FPSC X10 Files/Mapbank/Tutorials/Segments06.fpm"

HINT: Many times you will be working with segments and will want to make a modification to an existing room. You can quickly select a segment type by pressing the "K" key. This will put you into PICK mode. Just click on the cell with the segment type you are interested in and then you can start painting with it.

5.2.1 Segment Overlays

There are three types of segments that are classed as base segments. They are Rooms, Floors and Corridors. These segments are your starting blocks over which you can lay other segments like windows and doors etc.

Adding Stair Segments

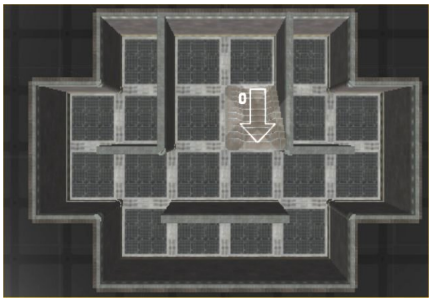
We are going to add some stairs to the level designed so far in this section. Make sure you are viewing Layer 5 of the level.

Click on the "add new segment" icon and point the browser to this path:

"Segment Library/Scifi/Platforms"

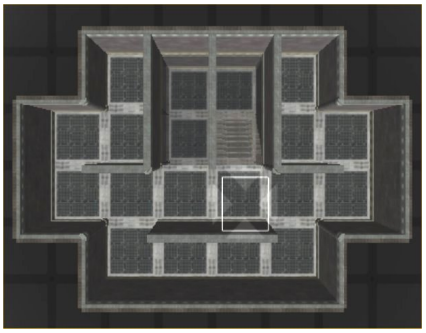
Choose the "Staircase metal" segment. The 3D cursor will show an arrow and also a small O (meaning Overlay).

Position the cursor as shown below and press the left mouse button to paste the stairs into layer 5.



Our room now has some stairs, move up to layer 6 (+ key). Press "K" to enter PICK mode, click on a cell that has the "Control Full Room" segment used.

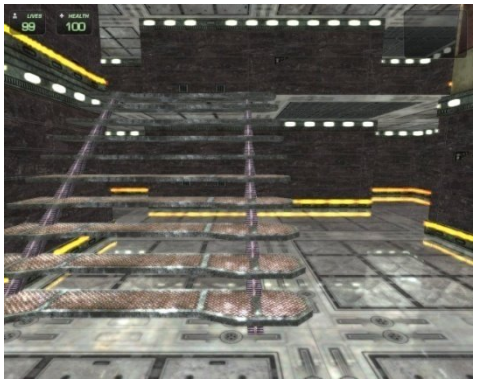
Press "F" to choose to only paint with Floors. Now paste some flooring as shown here:



🗁 "FPSC X10 Files/Mapbank/Tutorials/Segments07.fpm"

Internal and External Painting

The level editor helps you by automatically calculating the style needed to create the walls. Our level so far looks like this both from inside and out:

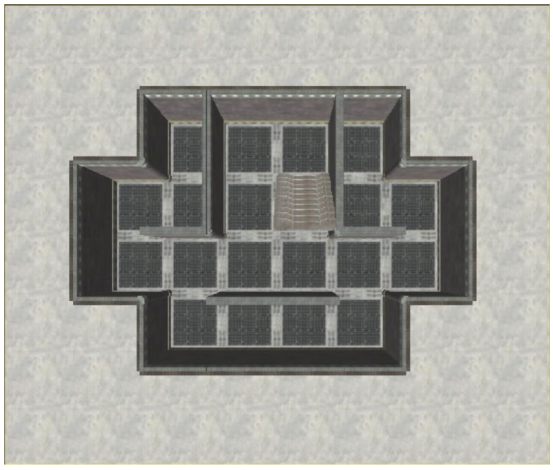


View inside the level.
Note the design of the walls.

View from outside the level. The outer walls are of a different exterior design

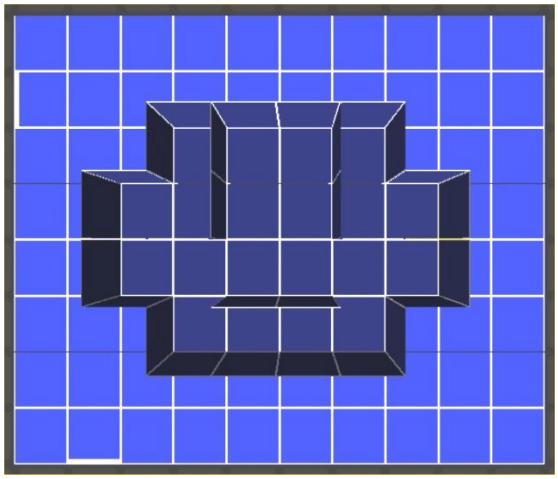
Let's now emulate the outside scene. Select a new segment, this time "Segment Library/Scifi/Platforms/Metal". This segment is a FLOOR only segment (it has no walls).

Ensure you are at Layer 5 and start painting the Metal segment around the outer walls, as shown here:



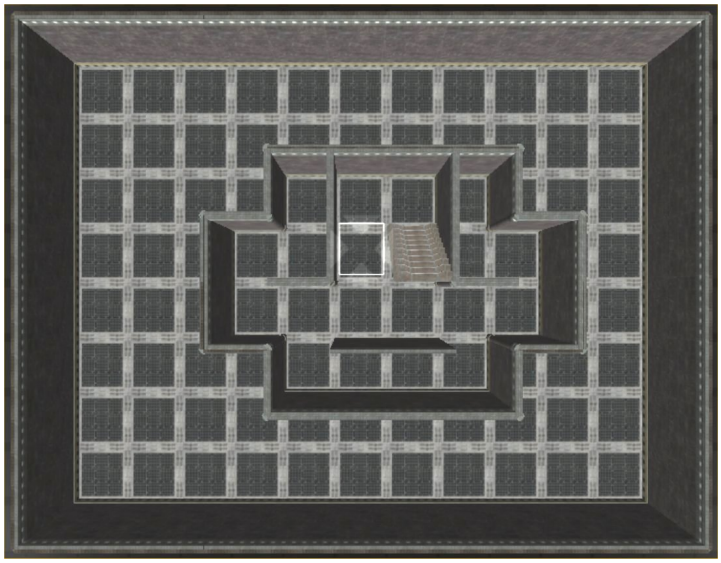
🗁 "FPSC X10 Files/Mapbank/Tutorials/Segments08.fpm"

If you decrease the render shroud (press the ` key) and you will reveal the blue print - notice that the cells are coloured differently:



The lighter blue coloured cells indicate external segments and the dark blue indicate internal segments.

You can use the external paint mode with the Control Room Full segment. This will help the editor determine how you want the wall structure defined. This next screen shot shows all the metal segments replaced with room segments but they have been drawn using the external paint mode:



🗁 "FPSC X10 Files/Files/Mapbank/Tutorials/Segments09.fpm"

You can see that the original internal walls are not disrupted when the outer segments were drawn. Of course you could have drawn the whole area with internal segments and then painted in the walls – there's more than one way to skin a cat!

Note: You cannot mix different component parts of segments with other segments. If you want to create such variations then you need to use the segment editor to build new segments. You can download this tool from the official FPS Creator X10 web site.

5.3 Entities

Entities add life to your game, any object that moves or has some reaction to the player's input is classed as an entity. Examples of entities are sliding doors, aggressive enemy robots, placid tea cups, collectable ammo, weapon pick ups, health bonuses etc.

Dynamic or Static?

Entities are very exciting when it comes to breathing life into your game. There is a trade-off here though as the use of entities creates more work for the game engine. This is because entities need to be checked each time the game loop is processed. So it's important you understand the types of entities there are and you become wise as to how best to employ them. An entity can be either **dynamic** or **static**. A dynamic entity is one that may move and react to things happening within the game. An example of a dynamic entity is a weapon. The weapon will be checked to see if the player has walked near to it. If the player has, then the weapon is picked up by the player and it's removed from the game scene. A static entity is one that never moves or reacts during the game. The normal use of static entities is the populating of your level with furniture.

An important point to realise is that if you make a level in single player mode and then switch it to multiplayer mode, all dynamic objects will be forced into static mode.
All of the supplied entity media in FPS Creator X10 has been set up as either dynamic or static. We have chosen the best state for each entity. You can easily change the state of the entity as we will show you.

To select an entity, click on the "Entity" tab at the base of the Library window and then click on the "Add new Entity" icon. Once an entity is selected and is following your mouse movements, you can paste copies of it into the game level by pressing the left mouse button.

In the screen shots below, we can see an entity called "generator". It has just been chosen as a new entity from the Library and is attached to the mouse. A red circle is underneath the entity and this signifies that it is designed as a static entity. By pressing the "Y" key, the entity can be changed to dynamic, upon which the red circle will change to green.

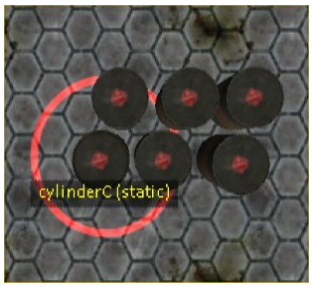
Static Entity
These cannot move when the game is running and cannot be destroyed or animate. On the plus side you can have many static objects without effecting the game's speed.

Dynamic entity
These entities are affected by the physics system. So you push them, pick them up, even blow them to pieces!

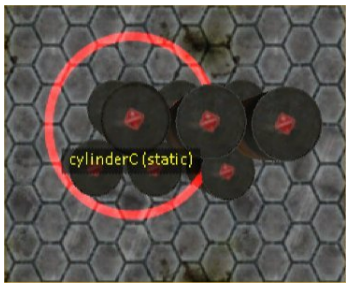
Stacking entities

Building a stack of entities can make for interesting level designs. By pressing the "Enter" key, the entities Y position (height) will be forced towards the floor. In most cases an entity will already be positioned onto the floor when you first select it. It is possible to move entities in all directions, so forcing the object to find the floor is very handy.

Sometimes other entities will be in the way of the object when finding the floor. This allows you to stack objects as this sequence describes:



First place down a few cylinders directly on the floor.

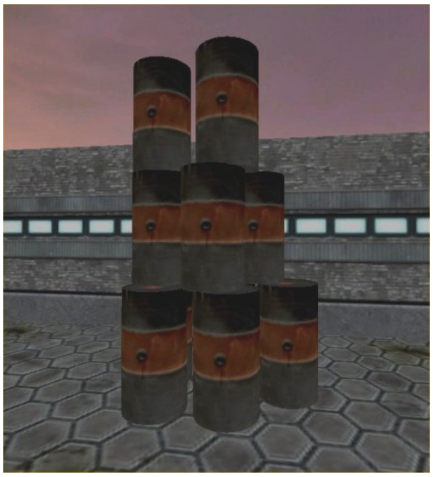


Press ENTER when over the barrels. This now sets the barrel on top of the lower barrels.

Repeat the process to build a third level of barrels.

🗁 "FPSC X10 Files/Mapbank/Tutorials/Entities_Stacked.fpm"

The resulting stack of barrels looks like this in the game:



In the demo file we have stacked barrels that explode as soon as you shoot them, a stack of barrels that are static and cannot be moved or exploded and a stack of barrels that take more than one shot to explode. This shows the flexibility of the static and dynamic modes and the tweaks you can make by editing the barrel's properties.

Key things to remember when you are using entities:

- Left mouse button pastes an entity
- Right mouse button deletes the currently held entity (so does the Delete key)
- Left click on an entity already in the level and it will assign itself to the mouse
- R key rotates the entity 90 degrees clockwise. Shift+R rotates the entity 1 degree clockwise for fine positioning. Ctrl+R rotates 10 degrees each time.
- Shift+Enter will force the entity down until it touches either another entity or the floor. This is useful if you want to stack item on shelves.
- As you stack objects higher, you may find that they seem to disappear from view when placed down. This is because the camera is now under the object you just placed. You will need to go up a layer to see where the object is positioned.

5.3.1 Entity Properties

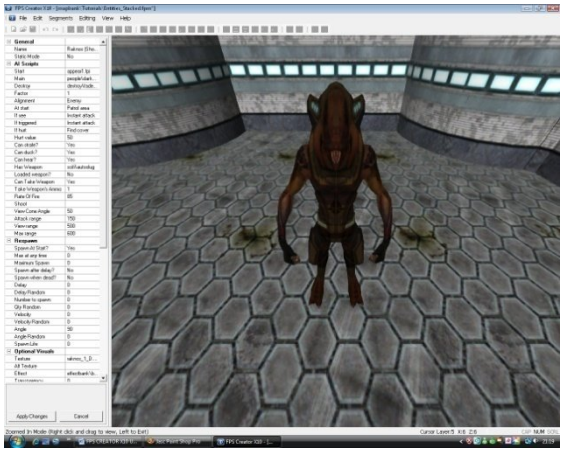
All entities have properties. Properties are information that control how the entity will perform various tasks. For example a door entity would have many properties from the speed the door opens and closes, to the sound the door makes when it is used. You can change the sounds for the opening and closing of the doors by editing the Sound0 and Sound1 properties of the entity. Sound0 is the opening sound and Sound1 is the closing sound.

The enemy entities have a whole host of properties. Changing these will allow you to customise your game in such a way that it becomes unique and different to the default settings.

🗁 "FPSC X10 Files/Mapbank/Tutorials/Entity_Properties.fpm"

1. In the centre of the level is a character entity. Make sure you are in Entity mode by pressing the "N" key.

Right click on the character. The 3D window will zoom down to closely inspect the entity and the properties will be displayed in a column on the left:



2. Locate the property labelled as "Health" and change the value from 100 to just 5. Now the entity will be easy to kill.
3. Change the property "Speed" from 100 to 200. This will make the entity run twice as fast as they normally would.
4. Click on Apply Changes" to approve of these modifications.
5. Click on "Test Game" and see how the enemy's attack has been customised.

Zoom Mode Controls
When you right click on an entity and enter the property editing mode, there are controls you can use to view and edit:

Right mouse button + drag mouse	When holding down the right mouse button you can spin around the selected entity to view it from different angles.
Mouse wheel	If you have a mouse wheel you can use it to zoom in and out from the entity.
Left & Right Arrow Keys	Moves the entity left or right relative to the camera's position.
Up & Down Arrow Keys	Moves the entity away and towards camera.
Page Up/Page Down	Moves the entity up and down the Y axis.
1 & 2	Rotates the entity around the Z axis
3 & 4	Rotates the entity around the Y axis
5 & 6	Rotates the entity around the X axis
Enter	Forces the object to "find the floor". Objects designed to hang on walls will find the nearest wall (they must be close to the wall for this to work).

5.3.2 Entity AI

All entities are controlled by Artificial Intelligence (AI) scripts. A script is a program that tells the entity what to do. You don't need to understand how the scripts work to use FPS Creator X10 – you can just change which script an entity uses and the behaviour of the entity will change in the game.

If you like to write computer programs, then it's likely you'll want to code your own scripts to show off how cool you can make your entities. In the "How To" section later on you will find details about the script language structure and its commands.

The properties of entities provide you with the means to customise your games endlessly. We have provided weapons, ammo, characters and other entities with properties that once changed will make the entity behave differently. You can take any entity and tweak it to ensure your game has your own definitive mark.

It can be great fun, imagine making some super grenades that you can throw as fast as a missile and that explode when they have bounced three times – ideal for taking out enemies around corners!

Below we have listed each entity type with all possible properties along with helpful description as to how they affect the entity.

In some cases you will find that similar entities don't share the same properties. For example, you will not find a "Throw Angle" property in the machine gun's list, but you will find it in the grenade's list.

5.3.3 Weapon Entities

Weapons are what the player will either start the game holding or pick up as they progress through your game. It's important to balance your game, so make sure you don't under arm your players (make it too difficult) and conversely, don't over arm them (make it too easy).

The properties listed below do not all apply to each weapon. For example, with projectile weapons you will have lifespan, throw speed, throw vertical, bounce quantity and explode on hit – but these do not apply to weapons that shoot.

General

Name	tavor	The in-game name for the entity, often used to reference the entity when triggering or activating entity behaviour.
Static Mode	No	The state describing whether the entity is dynamic within the game or part of the static level geometry.

AI Scripts

Start	Appear1.fpi	The FPI file that is run when the entity is first created within the game level.
Main	Weapon1.fpi	The FPI file that is run during the life of the entity.
Destroy	Disappear.fpi	The FPI file that is run when the entity is destroyed, meaning that the entity's health has reached zero.
Factor	1	Certain actions within the FPI script will use the FACTOR value, typically to increase and decrease the rate of change
Damage	5	Sets the amount of damage the weapon will do, zero meaning no damage at all.
Accuracy	10	Sets the accuracy of the weapon being fired, with zero meaning deadly accurate and 100 being wildly inaccurate.
Impact Force	0	Sets the force that the weapon will apply to the object it hits, zero being no force and 100 being full force
Reload Quantity	1	Sets the number of bullets allowed in one clip of the weapons ammo (if applicable), which can range from single shells represented by 1 to a magazine of 120 bullets per machine gun clip.
Fire Iterations	0	Sets the number of additional iterations describing how many extra streams of fire per shot. A value of 2 will shoot three damaging streams from the gun in one shot.
Lifespan	5000	Sets the number of cycles the projectile (if applicable) will survive before automatically exploding. The longer the lifespan the further a bullet could travel, ideal for sniper weapons.
Throw Speed	1	Sets the speed of the projectile (if applicable), a low value being slow and a high value being fast.
Throw Vertical	0	Sets the angle which the projectile will initially be launched (if applicable) from the launcher.
Bounce Quantity	0	Sets the number of times the projectile (if applicable) will bounce before exploding. A value of zero means the projectile does not explode due to bouncing.
Explode On Hit	No	Sets whether the projectile (if applicable) will explode on contact with a solid surface, "No" means the projectile will not explode on contact.

Respawn

Spawn At Start	Yes	If set to YES the entity spawns immediately, otherwise it waits to be activated

Max at any time	1	Sets whether the entity will spawn itself after being destroyed. A value of zero means it will not. A positive value will determine how many entities can be spawned of this type at any one time during the game.
Maximum Spawn	15	This sets the maximum amount of times this entity can be spawned. When this value is reached no more entities will spawn.
Spawn after delay?	Yes	If set to Yes, new entities of this type will respawn when the delay counter has counted down. A setting of No will ignore the delay parameter.
Spawn when dead?	Yes	If this is set to Yes then the re-spawn will happen after the entity is destroyed.
Delay	3000	Sets the delay before the entity will spawn again after being destroyed. The higher the value, the longer it takes to respawn. Zero means instantly respawn. A single unit represents a logic cycle. There are thirty logic cycles per second for entities near to the player, and much fewer cycles for entities further away.
Delay Random	0	Adds a random amount of time between 0 and this value to the Delay factor
Number to spawn	1	Sets the number of entities that will be respawned at the spawn event. Useful for spawning several clips of ammo or a nest of enemies in one go.
Qty Random	0	Adds a random quantity between 0 and this value to the number of entities spawned
Velocity	0	Sets the velocity at which the entity will be travelling when spawned
Velocity Random	0	Adds a random velocity between 0 and this value to the initial entity velocity
Angle	90	Sets the angle of elevation the entity will be pointing when spawned
Angle Random	0	Adds a random angle between 0 and this value to the initial angle of elevation
Spawn Life	0	Sets the amount of time in seconds the entity will live for. A value of zero means the entity will not die due to time

Optional visuals

Texture		The primary texture the entity uses to paint itself.
Alt Texture		The alternative texture initially hidden until an FPI script tells the entity to change to the alternative texture.
Transparency	1	A value of zero means the entity will not be semi-transparent. A value of one means the entity will use any alpha data the entity texture contains.
Reduce Texture	-1	A value of zero means the entity texture is unaffected and will use the game defaults. A value of minus one means no texture reduction will take place on the entity, allowing the entity to reveal maximum visual detail at the cost of memory consumption.
Texture Array	1	A value from 1 to 5 will set the texture variation of the entity to one of the built-in texture array choices, altering the entities appearance. A value of zero means a random variation will be chosen at runtime.
Refraction Pattern	Random	This field is used when the entity is using refraction, such as window entities. A drop down menu will allow a choice of refraction patterns to be selected, which will be used to shape the surface of the glass and distort the view beyond.

Physics

Physics on	YES	The state describing if the entity should use real physics, or revert to using no physics. Useful for dynamic entities that do not move, where you wish to speed up your game in certain areas.
Always Active?	No	If set to YES the entity will always be active, no matter how far away the player might be. Useful when you want to ensure characters always look intelligent no matter where they are in relation to the player, or keep entities stacked or spawned from the very start of the game. Remember that AI time is critical, so don't have too many entities using this.
Physics Weight	5000	This value controls the actual weight of the entity and will affect how it behaves in the game. A heavy entity will be hard to push, impossible to pick up and move very slowly when

		being tilted. A light entity will bounce delicately, be easy to pick up and respond to any effects around it, no matter how subtle.
Physics Friction	5000	This value controls the rough nature of the surface of the entity. A small value will mean the entity is smooth and will slide easily across any surface, whereas a high value will mean the entity is hard and course, meaning it will resist being dragged or pushed across the floor.
Physics Force Damage	100	Sets the amount of damage this entity will deal out if it hits the player or other entity
Rotate If Thrown	1	When set, this will rotate the object if the player picks it up and throws it. Useful for more realism when throwing bottles and cans, though other objects such as crates and square shapes may benefit from not rotating if the game play requires that they be stacked.
Explodable?	NO	If set to YES, the entity will explode when its strength reaches zero
Explode Damage	100	Sets the range and amount of damage this entity deals when it explodes

Ammo

Quantity	1	Sets the number of clips the ammo entity represents. Not to be confused with individual bullets, which a clip can comprise a number of. Bullet compliments of weapon clips are fixed to the weapon you chose.

5.3.4 Ammo Entities

Having a weapon isn't much use unless it comes with Ammo. Strategically placing ammo in hidden areas around your game will add to the game play experience and should ensure the user doesn't just walk through shooting and not being challenged. Be careful, don't hide all ammo – keep a balance, offer some in obvious places and reward players when they find a hidden stash.

From this point on, the respawn and physics property blocks will not be shown in full, so just refer to them above in the weapon properties list.

General

Name	assaultrifle_ammo	The in-game name for the entity.
Static Mode	No	The state describing whether the entity is dynamic within the game or part of the static level geometry.

AI Scripts

Start	Appear1.fpi	The FPI file that is run when the entity is first created within the game level.
Main	Ammo1.fpi	The FPI file that is run during the life of the entity.
Destroy	Disappear.fpi	The FPI file that is run when the entity is destroyed.
Factor	1	Certain actions within the FPI script will use the FACTOR value, typically to increase and decrease the rate of change

Respawn (see weapon properties)

Optional visuals

Texture		The primary texture the entity uses to paint itself.
Alt Texture	Ammo_D2.tga	The alternative texture initially hidden until an FPI script tells the entity to change to the alternative texture.
Effect		Select an .FX shader file
Transparency	1	A value of zero means the entity will not be semi-transparent. A value of one means the entity will use any alpha data the entity texture is using.
Reduce Texture	-1	A value of zero means the entity texture is unaffected and will use the game defaults. A value of minus one means no texture reduction will take place on the entity, allowing the entity to reveal maximum visual detail at the cost of memory consumption.
Texture Array	1	A value from 1 to 5 will set the texture variation of the entity to one of the built-in texture array choices,

		altering the entities appearance. A value of zero means a random variation will be chosen at runtime.
Refraction Pattern	Random	This field is used when the entity is using refraction, such as window entities. A drop down menu will allow a choice of refraction patterns to be selected, which will be used to shape the surface of the glass and distort the view beyond.

Physics (see weapon properties)

Ammo

Quantity	1	Sets the number of ammo clips this entity represents.

5.3.5 Enemy Entities

We advise you to introduce enemies gradually, players don't want to load up a game and be blasted to death in seconds. They also don't want to walk for ages without seeing sight of any foes. There are plenty of properties you can tweak to make your enemies unique and different. You can also create your own scripts – this will certainly put your own stamp of creativity on your game.

General

Name	conker (grenade)	The in-game name for the entity.
Static Mode	No	The state describing whether the entity is dynamic within the game or part of the static level geometry.

AI Scripts

Start	Appear1.fpi	The FPI file that is run when the entity is first created within the game level.
Main	Ammo1.fpi	The FPI file that is run during the life of the entity.
Destroy	Disappear.fpi	The FPI file that is run when the entity is destroyed, meaning that the entities health has reached zero.
Factor	1	Certain actions within the FPI script will use the FACTOR value, typically to increase and decrease the rate of change
Alignment	Enemy	Sets whether the character is an enemy or friend, and neutral means they are not involved
At start	Patrol area	Sets the behaviour when the character is first created
If see	Instant attack	Sets the behaviour when the character sees their target
If triggered	Instant attack	Sets the behaviour when the character is activated by a trigger event
If hurt	Find cover	Sets the behaviour when the character is hurt past the hurt value
Hurt value	50	Sets the threshold below which the character is considered to be hurt
Can strafe?	Yes	Sets whether the character can strafe side to side or not
Can duck?	Yes	Sets whether the character can duck or not during combat
Can hear?	Yes	Sets whether the character can hear any sound made in the game
Has weapon	Scifi/glaive	Sets the name of the weapon the enemy is holding, if any. Each character has a default weapon, and this can be changed by selecting a new weapon in this field.
Loaded weapon?	No	Sets whether the characters weapon is loaded when they are created
Can Take Weapon	YES	Sets whether the player can take the weapon if any when the entity enemy dies, with a YES indicating the player can take the weapon.
Take Weapon's Ammo	1	Sets the number of clips the player will collect along with the weapon if the player collects the weapon from the dead entity.
Rate Of Fire	20	Sets the speed at which the enemy fires the weapon at the player, 0 meaning a constant rate and 100 being sporadic bursts from time to time.
Shoot	People/hunt-shoot...	Sets the FPI filename to be used when the entity issues the USEWEAPON action in the FPI script. Typically triggered when the enemy sights the player and fires.
View Cone Angle	50	Sets the horizontal field of view of the entity, used when the entity is asked to detect the player based on sight. A low value will give the enemy tunnel vision. A larger

		value will provide the enemy with great sight awareness.
Attack range	150	Sets how close the character should get to the target
View range	500	Sets the range at which the character will be able to see their target
Max range	600	Sets the maximum range beyond which the character cannot hear or be aware of others

Respawn (see weapon properties)

Optional visuals

Texture		The primary texture the entity uses to paint itself.
Alt Texture	Conker_d2.tga	The alternative texture initially hidden until an FPI script tells the entity to change to the alternative texture.
Effect		Select an .FX shader file
Transparency	0	A value of zero means the entity will not be semi-transparent. A value of one means the entity will use any alpha data the entity texture is using.
Reduce Texture	0	A value of zero means the entity texture is unaffected and will use the game defaults. A value of minus one means no texture reduction will take place on the entity, allowing the entity to reveal maximum visual detail at the cost of memory consumption.
Texture Array	1	A value from 1 to 5 will set the texture variation of the entity to one of the built-in texture array choices, altering the entities appearance. A value of zero means a random variation will be chosen at runtime.
Refraction Pattern	Random	This field is used when the entity is using refraction, such as window entities. A drop down menu will allow a choice of refraction patterns to be selected, which will be used to shape the surface of the glass and distort the view beyond.
Show health?	Yes	Sets whether a health bar is displayed for the character or not
Show ally icon?	Yes	Sets whether the follow, hold and attack commands issued by the player to allies triggers an icon above the heads of all valid allies.

Statistics

Lives	1	Sets the number of lives the entity starts the game with. This value ranges from 1 to as many lives as you want it to.
Health	25	Sets the initial health level of the entity, which can be any value greater than zero. If the value is zero, the entity is invulnerable to damage.
Speed	100	Sets the general speed of the entity, which can be used by the entity to control movement, animation or any process that occurs over time.
HurtFall	100	Sets the distance beyond which a fall will hurt the entity or player. A value of zero means any vertical landing will hurt, where a value of 200 means the player can safely drop two layers before it will hurt.
IsImmobile	No	No means the entity is mobile and can move around and is affected by forces. Yes means the entity is completely immobile and will never move.
IsObjective	0	This sets the level objective of the entity. A value of zero means this entity is not an objective, where a value greater than zero means if the entity is destroyed the objective number is complete. If two entities share an objective number, either entity can be used to complete that objective value. If two entities have different objective numbers, both entities must be destroyed/collected to complete the objectives of the level.
Score	10	Specifies how many points this entity is worth. The points are added to the players score if the entity is destroyed or collected.
Decay Time	10	Specifies in seconds how long the corpse of a fallen character will remain on the ground before disappearing. A value of zero will keep the corpse in the game forever.

Physics (see weapon properties)

Sound

Soundset	robotic	Specifies the WAV file that can be accompanied by this entity. The sound is triggered from certain FPI scripts.

Character AI

Thanks to the new DarkAI component, characters have an extended AI system which controls their behaviour during the game. The four actions available in the behaviour combos are:

NO ACTION

Performs no alternative action, and resorts to the characters natural reactions based on the condition. This normally is to let the automated AI take charge. In the case of friendly characters, this default setting will keep the ally stationary at their start location until the player is within their range.

INSTANT ATTACK

Forces the character into an aggressive posture, where they will attack their target immediately, ignoring what they where doing before. This action has no influence on neutral characters.

FIND COVER

Forces enemy characters to immediately take cover at the nearest available 'Enemy Safe Zone' which can be found in the markers section. A marker is available if no other character is using it. If there are no markers within range, the character returns to its previous behaviour. This action has no influence on friendly and neutral characters.

PATROL AREA

Instructs a character to begin scouting the local area for enemy targets, rather than stand at their idle position. The pattern of their patrol is random. This action has no influence on neutral characters.

5.3.6 General Entities

The majority of entities used when designing your levels are classed as "General Entities" Changing the texture or the sound of an entity will ensure a unique style for your game.

General

Name	torch	The in-game name for the entity.
Static Mode	No	The state describing whether the entity is dynamic within the game or part of the static level geometry.

AI Scripts

Start	Appear1.fpi	The FPI filename that is run when the entity is first created within the game level.
Main	Default.fpi	The FPI filename that is run during the life of the entity.
Destroy	Disappear.fpi	The FPI filename that is run when the entity is destroyed, meaning that the entity's health has reached zero.
Factor	1	Certain actions within the FPI script will use the FACTOR value, typically to increase and decrease the rate of change

AI Automated

Use Key		Holds the name of the entity the player needed to have collected in order to USE this entity, typically used by a door to indicate which key unlocks it.
If Used		Holds the name of the entity to be activated by the FPI script used by this entity, typically associated with a door script waiting for the player to perform the USE action.

Respawn (see weapon properties)

Optional visuals

Texture	Texturebank\scifi\ac...	The primary texture the entity uses to paint itself.
Alt Texture		The alternative texture initially hidden until an FPI script tells the entity to change to the alternative texture.
Effect		Select an .FX shader file
Transparency	0	A value of zero means the entity will not be semi-transparent. A value of one means the entity will use any alpha data the entity texture is using.
Reduce Texture	0	A value of zero means the entity texture is unaffected and will use the game defaults. A value of minus one means no texture reduction will take place on the entity, allowing the entity to reveal maximum visual detail at the cost of memory consumption.
Texture Array	1	A value from 1 to 5 will set the texture variation of the entity to one of the built-in texture array choices, altering the entities appearance. A value of zero means a random variation will be chosen at runtime.
Refraction Pattern	Random	This field is used when the entity is using refraction, such as window entities. A drop down menu will allow a choice of refraction patterns to be selected, which will be used to shape the surface of the glass and distort the view beyond.

Statistics

Strength	0	Sets the initial strength of the entity, which can be any value greater than zero. If the value is zero, the entity is invulnerable to damage.
Speed	100	Sets the general speed of the entity, which can be used by the entity to control movement, animation or any process that occurs over time.
IsImmobile	No	No means the entity is mobile and can move around and is affected by forces. Yes means the entity is completely immobile and will never move.
IsObjective	0	This sets the level objective of the entity. 0 = not an objective 1, 2, 3.. etc = Is an objective Only one entity with the same IsObjective value has to be completed. Entities with different values must all be completed.
Score	10	Specifies how many points this entity is worth. The points are added to the players score if the entity is destroyed or collected.

Physics (see weapon properties)

Sound

Sound0		Specifies a sound file that can be called up from one of the FPIs.
Sound1		Specifies a sound file that can be called up from one of the FPIs.

5.3.7 Description of the general FPI scripts

The FPI scripts that the default entities use in FPS Creator X10 have specific jobs to do. A one line filename is not ideal for describing what they do. Some FPI scripts link into an integrated component to handle character behaviours, called DarkAI, which allows more complex logic for allies, enemies and neutral characters which would be difficult to reproduce with FPI actions. In previous versions of FPS Creator, this logic was performed by a series of FPI scripts for character control. This behaviour is now controlled directly from the entity properties panel as described above and the FPI scripts DARKINIT.FPI and DARKAI.FPI. The purpose of this list is to explain them better, you can then decide whether they would be suitable for your current entity or not.

Destroy FPIs

Crumble.fpi	End script used to destroy an entity by playing a crumble animation from it

destroyandactivate.fpi	Destroys this entity and activates another as specified in the IFUSED field
Flammable.fpi	End script used to trigger an explosion from the entity when it is destroyed
Window.fpi	End script used to shatter a pane of glass in the location of the entity (ie glass)

Effects FPIs

incrementambience.fpi	Increment the ambient light level by the value specified in FACTOR
decrementambience.fpi	Decrement the ambient light level by the value specified in FACTOR
raisewaterlevel.fpi	Increment the water level to match the height stored in FACTOR
lowerwaterlevel.fpi	Decrement the water level to match the height stored in FACTOR
Incrementbloomsize.fpi	Increment the bloom size by the value specified in FACTOR
Decrementbloomsize.fpi	Decrement the bloom size by the value specified in FACTOR
Incrementbloomsize.fpi	Increment the bloom strength by the value specified in FACTOR
Decrementbloomstr.fpi	Decrement the bloom strength by the value specified in FACTOR
Incrementmusic.fpi	Increment the music volume by the value specified in FACTOR
Decrementmusic.fpi	Decrement the music volume by the value specified in FACTOR
Incrementsounds.fpi	Increment the sound volume by the value specified in FACTOR
Decrementsounds.fpi	Decrement the sound volume by the value specified in FACTOR
Incrementwatersmoothness.fpi	Increment the water smoothness by the value specified in FACTOR
Decrementwatersmoothness.fpi	Decrement the water smoothness by the value specified in FACTOR
Incrementwaterspeed.fpi	Increment the water speed by the value specified in FACTOR
Decrementwaterspeed.fpi	Decrement the water speed by the value specified in FACTOR

General FPIs

Ammo.fpi	Main script assigned to items that are to be picked up and used as ammo
Ammoglow.fpi	Identical to ammo.fpi script except that it will create a glow effect decal
Appear1.fpi	Initialisation script used to make the entity appear instantly
Appear2.fpi	Initialisation script used to make the entity fade into the game slowly
Appearnogravity.fpi	Use this to initialise the entity so that it will not obey the normal laws of physics gravity, and remain neutral to its force which is good for floating platforms
Appearspawn.fpi	This will cause the entity not to leave the appear script until the entity has been activated, upon which it will activate the spawn feature of the entity.
Autodoor.fpi	Main script used to open an animating door when the player or entity gets close
Controlspawn.fpi	This will activate entity spawning when the entity itself is activated, and switch off spawning when the entity itself is de-activated
Decal.fpi	Used to control the emission of a decal from an invisible entity marker
Decalfixed.fpi	Identical to decal.fpi except that the decal will not rotate to face the player
Decalflat.fpi	Identical to decal.fpi except that the decal faces upwards though flat on the floor
Default.fpi	An empty script to be used when nothing should be done with the entity
Disappear1.fpi	End script used to destroy an entity instantly
Disappear2.fpi	End script used to destroy the entity by fading it out of the game slowly
Door1.fpi	Main script used to open and close a door by player proximity only
Doorkey.fpi	Main script used to open a door only when the player has collected the right key
Doorremote.fpi	Main script used to open a door when it has been activated from a remote switch
Dooruse.fpi	Main script used to open a door when the player performs the USE action on it
Emission.fpi	Main script used to create looping decals from an invisible entity position
Followнorotate.fpi	Causes an entity to follow a waypoint, and keep its rotation fixed so it will not turn and face its movement direction, useful for floating platforms.
Healthuse.fpi	Main script used to provide the player with health if the entity is USED
Lift1.fpi	Main script used to automatically move the lift up and down if player stands on it. Activating entity brings lift to players present layer.
Lift2.fpi	Main script used to automatically move the lift up and down if player stands on it
Light1.fpi	Main script used to control a dynamic light, default is lit, activated switches it off. The second variant flickers the light on and off randomly.
Light2.fpi	Main script used to flickers the light on and off randomly.
Lightoff.fpi	Main script used to switch the light off when the level starts.
Musicinzone.fpi	Will play the music file specified in SOUND0 field for as long as the player is inside the zone associated with this entity.
Pickup1.fpi	Main script used by items that are to be picked up during the game
Pickuphealth.fpi	Identical to pickup1.fpi except that in addition, the player receives health points

Pickuphealthglow.fpi	Identical to Pickuphealth.fpi except that in addition, the entity will have a glow beneath and the entity will float up and down, while spinning
Pickupkey.fpi	Main script used by keys that are to be picked up during the game
Pickupoxygen.fpi	Main script used by entities that will provide air to the player
Plrhealzone.fpi	Main script to give the player health if they are standing in the associated zone
Plrhurtzone.fpi	Main script to take health from player if they are standing in the associated zone
Plrinzone.fpi	Main script which will activate a named entity if the player enters the zone
Plrinzoneactivateused.fpi	Similar to PLRINZONE.FPI but will only activate entities with the same name as that specified in the IFUSED field of the entity associated with this FPI script
Repeatsound.fpi	Initialisation script used to trigger a repeating sound from an entity
Soundinzone.fpi	Will play a sound once when the player enters the associated zone
Soundloopinzone.fpi	Similar to SOUNDINZONE.FPI, but will loop the sound continually instead of a single playback
Storyinzone.fpi	Will play a video once when the player enters the associated zone
Switch1.fpi	Main script to activate an assigned entity if player USES it, close proximity
Switch2.fpi	Similar to Switch1.fpi but uses texture toggle instead of animation
Switch3.fpi	Identical to switch2.fpi except that this script is used to control animating switch
Switch.fpi	Identical to switch2.fpi except that this switch can be used from further away
Transportifused.fpi	Main script used to teleport the player to the entity named in IFUSED field
Transporttoexit.fpi	Main script used to teleport the player to the Teleport OUT entity only
Weapon.fpi	Main script used to handle an entity to be picked up and used as a weapon
Weaponglow.fpi	Identical to weapon.fpi except that a decal glow is added to the entity
Zoneactivate.fpi	Main script used to activate ALL entities within zone when player stands in zone
Zoneanyactivate.fpi	If any entity enters the zone associated with this entity, it will activate the entity named in the IF USED field and play the sound specified in the SOUND0 field
Zoneanykeyactivate.fpi	If any 'key entity' enters the zone associated with this entity, it will activate the entity named in the IF USED field and play the sound specified in the SOUND0 field

5.3.8 How Doors and Switches Work

All Doors use an FPI script to control how they work. Some FPI scripts are simple and others require explanation. Here is a run down of terms you will encounter:

FAKE	This means the door is no more than scenery and cannot be opened.
KEY	This means the door requires a key to be held by the player, and that the player needs to perform the USE action on the door to open it. The kind of key depends on the name that is written in the entity properties for the door, under the field name USE KEY. By default the key name is simply 'Key', and all keys that are added to the game are called 'Key'. You can increase the number of keys and doors they work on by giving them different names in the entity properties dialog, such as "BunkerKey", "DoorKey, "SafeKey", etc.
AUTO	This means the door will open automatically when the player or another entity gets too close to the door. There are two FPI scripts you can use for this type of door. The DOOR1.FPI which means ONLY the player can open the door, and the AUTODOOR.FPI which all other characters and the player can open automatically.
REMOTE	This means the door is closed and locked until it has been remotely unlocked, which is done using a SWITCH. Once the door has been unlocked, it will behave just like an AUTO door. All remote doors share the name 'Remote Door', and all switches by default will activate any entity called 'Remote Door', except light switches which will activate entities called 'Light'. In the case of remote switches for doors, the term to activate means to nudge it into action, and the action in the case of a door is to unlock.

As mentioned, in the case of KEY and REMOTE you also need a secondary entity to complete the logic, which are described here:

KEY	This is an entity you can find in the entity items area. Simply drag and drop a key into your level. The player can then find and pick it up (just by moving close to it), then when the player USES the key on the door, the door will unlock and open. The USE key is set to "Enter" on the keyboard by default.
SWITCH	This is an entity typically found in WALL FURNITURE which can be added to walls, and when USED by the player will activate any entities in your level called 'Remote Door'.

	You can change this in the IF USED field of the entity.
LIGHT SWITCH	This is an entity almost identical to SWITCH, except that it will activate any entity in your level called 'Light'. Note that all entity names can be changed.

5.3.9 Entity Explosions and Debris

Entities can be set up to detonate upon their destruction, causing damage to the surrounding area. This damage will apply to other entities in range and the player. The damage is only dealt if there is a direct line of sight between the exploding entity and the target. This feature is activated when the EXPLODING field is set in the entity properties of the dynamic entity of your choice.

Some entities can produce debris, the fragments of the entity that make up the object. There are usually 3 to 5 fragments ejected from the destroyed entity, and add a touch of realism to the entity being destroyed. These fragment objects play no part in the game play, though they can be pushed around by the player for a short time before they disappear. There is no property to control this debris feature, simply the choice of using an entity that does or does not exhibit the feature.

You can have a lot of fun by setting up objects that explode near enemies, allowing the player to take out the enemies by them firing at things like exploding barrels. Play around with values like the strength of an object and the explode damage property.

5.3.10 The FACTOR setting

The FACTOR value is a special variable which can be used internally by some FPI actions and conditions. By default this value is one, which means selected FPI actions would scale their effect by one, meaning no change in default behaviour. A value greater than one will scale the effect of the FPI action proportional to the new value, so for example if the FPI action was to increase the ambient light level to 100, the FACTOR value would determine how fast this transitional effect would take. A value of one would increment the light level in steps of one unit, the slowest transition, whereas a value of 10 would step the ambient light level in leaps of 10, making the transition very fast indeed. The FACTOR value is also used in other ways such as describing a destination value, such as the new height for the water level, where the FPI action will raise or lower the water until the value in FACTOR is reached. Whether an FPI action uses the FACTOR value or not is specified in the description of each FPI action elsewhere in this manual."

5.3.11 Dark AI scripts

The DarkAI.FPI and DarkINIT.FPI scripts are used for the absolute control of in-game characters. These scripts link directly into the internal AI system driven by the DarkAI module which is part of the engine. You do not need to modify these scripts as they have been specially designed to perform all the actions the character requires for neutral, enemy and friend behaviour as described elsewhere in the manual. If you do not want your enemy character to perform an initial spawn animation when they see the player, replace the DarkINIT.FPI with the APPEAR1.FPI, replacing the introduction of the character with simple appearance logic.

5.4 Markers

These types of game objects are for setting the start position of the player, rigging the lighting of the level and creating areas within the level that can create event based actions.

Types of Markers

Player Start

Represents where in the level the player will be created at the start. Use the "R" key to rotate the position. Each level of your game should have one of these. If FPSC does not find the marker you will end up being placed at an arbitrary position.

Player Checkpoint

Place these throughout your game levels. They dictate where the player will be re-spawned if the player dies. You should reward players when they have navigated a difficult section of a level by giving them a new check point. The player simply walks over the cell where the checkpoint is placed – the game then knows this is where the player will be returned to should they be killed. The direction of the arrow is also used, so you can design what the player sees when they are reborn.

White Light

This marks where a light source will be generated within the game's level. There are many variables associated with lights – make sure you read up on the pros and cons of the different light settings.

When you first select a light from the library it will be surrounded by a circle. This indicates the range that the light will reach. You can change the size of the range by pressing the "[" and "]" keys. The lights range of influence will expand or shrink depending on the key you press.

Although some basic colour choices are provided in the library, you can easily change the colour of the light by editing the light's properties.

Heal Zone

An area of the level that has been defined for a certain purpose. The most basic sub-type is the trigger zone. This has been set-up using an AI script called "plrinzone.fpi". The AI script checks to see if the player has entered the specified zone, if the player has then another entity can react, maybe turning lights on or opening a secret door.

FPSC comes with pre-defined zone markers: Heal, Hurt and Win. If the player walks into the Heal zone then the health of the player will increase. Walking into the Win zone will trigger the completion of the current level.

Sound Zone

This zone marker can be used to trigger off sound files. As the player enters a new room you could play a sinister piece of music. Or maybe when the player meets a friendly character a piece of information could be spoken that gives clues to where better weapons and ammo can be found.

Story Zone

An area of the level that when the player enters the specified zone, an item of media is played to support the game story. This item of media could be an AVI animation sequence, or a bitmap displayed on screen, and/or a WAV/OGG file played to provide a narrative. These zones are triggered only once when the player enters them for the first time.

Music Zone

An area of the level that when the player enters the specified zone, music is played to support emotional context within the level. This music is specified as a WAV/OGG file in the properties. These zones are triggered only when the player is inside the zone. The music ends when the player leaves.

Place these throughout your game levels. They dictate where the enemy will run to when they are instructed to FIND COVER as part of their overall behaviour. You should place these in your level so that as the player progresses forward, enemies have places to retreat to and hide should they require greater cover, such as crates or over-turned obstacles. The direction of the arrow is also used, so you can instruct the enemy in which direction they should look when they arrive at their safe spot.

5.4.1 Player Start Position Marker

General

Name	Player Start	The in-game name for the entity.
Has weapon		Sets the name of the weapon the player is holding at the start of the game (if any).
Loaded weapon?	No	Sets whether the players weapon is loaded when they first start the level.
Ammo Quantity	1	Specifies a quantity of ammo clips the player starts with.

Statistics

Lives	3	Sets the number of lives the player starts the game with. This value ranges from 1 to as many lives as you want. Most games offer 3 lives but with options to top-up health frequently scattered about.
Health	500	Sets the initial health level of the player, which can be any value greater than zero. If the value is zero, the player is invulnerable to damage.
Speed	100	Sets the general speed of the player, which can be used by the player to control movement, animation or any process that occurs over time. Don't set this too high, otherwise the physics system will become unstable.
HurtFall	100	Sets the distance beyond which a fall will hurt the player. A value of zero means any vertical landing will hurt, where a value of 200 means the player can safely drop two layers before it will hurt.
Jump Height	50	Sets the maximum jump height for the player. A value of 50 is the default height, whereas a value of 100 will allow the player to jump much higher and a value of zero means the player cannot jump.
Air capacity	500	Sets the amount of air the player can hold in their lungs before damage is taken.
Score Goal	0	Set this if you want your level to have an overall objective of points the player must win to complete the level and go to the next one or win the game. By default, the value of zero means the score objective is not used. A value greater than zero will activate the ability of the player to win the level by scoring the specified points.

Sound

Soundset	player	Specifies the WAV file that can be accompanied by this entity. The sound is triggered from certain FPI scripts. In the case of player start marker and characters, this field specifies the sound folder that holds a range of human and robotic noises.

5.4.2 Player Checkpoint Marker

General

Name	Player Checkpoint	The in-game name for the entity.

Sound

Sound0		Specifies either a WAV or OGG file that can be accompanied by this entity. In this case, the sound that indicates the player has reached a check point in the level.

5.4.3 Light Markers

As a default there are no lights in an FPSC level, you have to add them. In order to make the learning curve as gentle as possible we have added artificial ambient light as the default illumination when there are no user created lights to illuminate the scene. This ambient light is set bright so you can see everything clearly in full colour, however it offers no opportunity for mood or shadows.

There are two types of light you can add to your level, static and dynamic lights. Static lights are lights that calculate how your level should be illuminated and are then burned into your textures for the final game (the process known as light-mapping). This means any lighting you have created with static lights are permanent and cannot be changed during the game. Dynamic lights can be changed during the course of the game, and can be used to add lights which can switch on and off, such as a room with a light switch. Dynamic lights do not cast shadows on objects when light-mapping is performed.

All added lights default to static because they are the most performance friendly of the two choices and will give you a faster game. Dynamic lights take up extra processing and too many of them in any one area will slow down your game.

Dynamic lights are also constrained by using a fixed function pipeline which means there are only 6 actual hardware lights at any one time, and the impression of more than 6 lights is worked out by averaging colour and light positions to best effect. Naturally this may create visual curiosities when many dynamic lights are close together. It is suggested you keep your dynamic lights spread far apart and never have more than six dynamic lights that overlap their light range significantly or you will notice the averaging system change the light play as you walk through the lighted area to an 'unrealistic' extent.

Lights can be changed in terms of their colour and the range they can cast light, with a number of default choices already provided in the MARKERS section of the library. You can change the range and colour of the light by right clicking on any light you have added, and modifying the fields in the properties panel that pops up. You can also visually change the range of the light using the shortcut keys [and].

Light Switches
To create a room with a light and a light switch, you have to create a room, drop in a light and set that light to be dynamic. You can do this by simply pressing the Y key which toggles the light state away from static. Then drop in a light switch from the "entities\wall furniture" selection and place the switch on any wall surface. Now run the test game, go up to the switch and press the ENTER key to begin switching on and off. You will notice the dynamic light automatically switches on and off in response. The switch was automatically linked to the light by way of the entity names.

To create more switches and lights, you need to change two things: The name of the extra dynamic lights and the field within the switch properties that control which entity the switch affects. A good naming convention will keep track of all your switches and all your lights (i.e. Hallway Light Switch -> IFUSED=Hallway Light).

🗁 "FPSC X10 Files/Mapbank/Tutorials/Dynamic and Static Lights.fpm"

The only cost to having many static lights is the amount of extra texture memory used to store all the level lighting changes. Static lights represent the best way to atmospherically light your levels. We suggest you use dynamic lights only when they are a key element of the gameplay.

Here are the properties of a light marker:

General

Name	Light (White)	The in-game name for the light entity.
Static Mode	Yes	Defines if the light is a static or dynamic light.

AI Scripts

Start	Appear1.fpi	The FPI filename that is run when the entity is first created within the game level.
Main	Light1.fpi	The FPI filename that is run during the life of the entity.
Destroy	Disappear.fpi	The FPI filename that is run when the entity is destroyed, meaning that the entities health has reached zero. Lights can be destroyed in this way, and a destroyed dynamic light does not affect the scene.
Factor	1	Certain actions within the FPI script will use the FACTOR value, typically to increase and decrease the rate of change

Light

Light Range	500	Sets the range the light can illuminate. A value of zero means the light will not illuminate anything, and a value of 200 will illuminate two segment tiles in a spherical radius.
Light Color	[ ] ffffff	Sets the colour of the light using a colour picker, with which you can choose the colour of the illumination. The colour is also shown in hexadecimal format (which you don't need to understand).

5.4.4 Zone Markers

Zone markers are invisible to the player of the game but they provide you with a tool for setting up gaming events, such as the sound of an alarm triggering as the player enters a room, or you could spawn hidden enemies to attach the player.

Zones are easily placed and if you hold down the Shift key, you can stretch them over larger areas and even across layers.

The property list of zone markers is:

General

Name	Trigger Zone	The in-game name for the entity.

AI Scripts

Start	Appear1.fpi	The FPI file that is run when the entity is first created within the game level.
Main	plrinzone.fpi	The FPI file that is run during the life of the entity.
Destroy	Disappear.fpi	The FPI file that is run when the entity is destroyed, meaning that the entity's health has reached zero. A destroyed marker will no longer affect the level.
Factor	1	Certain actions within the FPI script will use the FACTOR value, typically to increase and decrease the rate of change
If Used	Remote Door	Name of the entity that will be activated when this zone marker is triggered off.

Statistics

IsObjective	0	This sets the level objective of the entity. 0 = not an objective 1, 2, 3.. etc = Is an objective Only one entity with the same IsObjective value has to be completed. Entities with different values must all be completed.

Sound

Sound0	Hello.wav	The sound played when the player enters the trigger zone.

The different zone markers that you can select from the library each call different fpi files during the game. So the hurt zone marker calls plrhurtinzone.fpi, the heal zone calls plrhealinzone.fpi and the win zone calls plrinzone.fpi. These fpi files directly control the health and state of the game.

5.4.5 Story Marker

This marker provides the tools for you to script a story to your single player games. As soon as the player enters a story zone, the game freezes and the selected video file or image file is displayed to the player. Pressing a mouse button or key will remove this image and the player continues the game.

A story zone has a media section in its properties.

Media

Audio	storytrigger.ogg	The sound played when the player enters the story zone.
Visual	storyvideo.avi	A video or image file that will be displayed as soon as the player enters the story zone area.

5.4.6 Stretching a marker over layers

It is possible to create a marker area that stretches over 2 or more layers. You might want to do this to create say a large hurt zone in a deep pit.

To achieve this you need to follow this control process:

1. Select and place a zone

2. Hold down the Shift key and Hold down left click over the marker icon

3. While left click is held down, release the shift key

4. While left click is held down move the mouse around

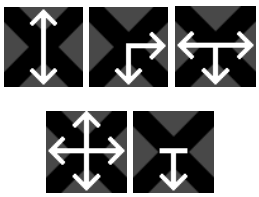
5. While left click is held down, press + and - to navigate up and down the layers

6. Release left click to complete the resizing of the zone

6 More Map Editing Information

6.1 Cursor Editing States

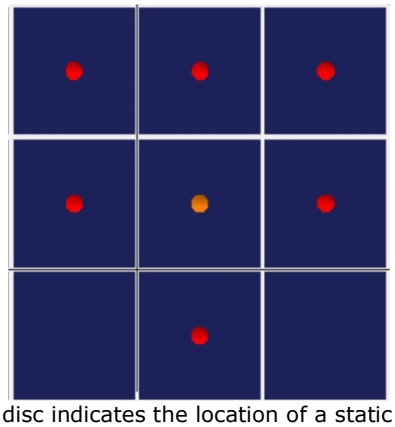
The 3D editor will change visually to help you identify the mode you are currently using. The full list is described here:

Left mouse button	Right mouse button	Images of 3D cursor	Key short cut
Paint segments	Delete segment		P
Paint only the Floor section of a segment	Delete Floor		F
Paint only wall section of a segment	Delete wall		W Use "R" to rotate arrow
Pick full segment from the map	n/a		K
Paint an overlay segment onto a full segment	Delete overlay		Use "R" to rotate overlay
Paint a segment overlay in the direction indicated by the arrows.	Delete segment		Use "R" to rotate overlay

- Note: If you are in "floor" or "wall" painting mode you can change back to painting with all floors and walls by pressing the "A"key.

6.2 Editor Indicators

As your levels increase in complexity it can be hard to remember where everything is placed. The editor provides some visual clues to help you locate where entities have been scattered around. If a segment contains one or more entities it will have a coloured disc in the centre when viewed in blue print mode, for example:



A red disc indicates the location of a static entity
and a green disc indicates the location of a dynamic entity.
A yellow disc indicates the location of a light marker

6.3 3D Clipping Modes

To see inside corridor segments, press the TAB key. This will hide all layers above and below the layer you are currently editing and it will also remove the top half of the layer being edited, revealing the inside of the corridor. This makes placement of entities and markers much easier.

There are three states to the TAB key:

State 1: View current layer and five layers below and above

Allows you to get a feel of how the current layer will align with any layers below or above it. The Editor has to render up to 11 layers at once, so you may only want to turn this on to see how you are doing and then revert back to viewing only the current layer.

State 2: View current layer only

Displays only the current layer. This state is the fastest for the editor to render.

State 3: View and clip current layer

Sometimes you will want to place entities into corridors. Because the corridors are self contained with their own ceiling, you will need this mode to cut off the ceiling to see where to place your entities.

6.4 Special Media

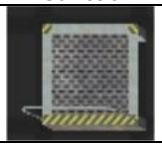
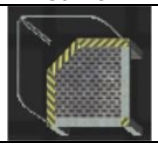
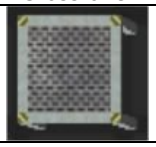
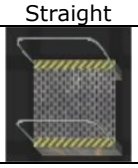
The media supplied with FPS Creator X10 includes some unique entities and segment overlays. We will look at these in detail here.

6.4.1 Gantries

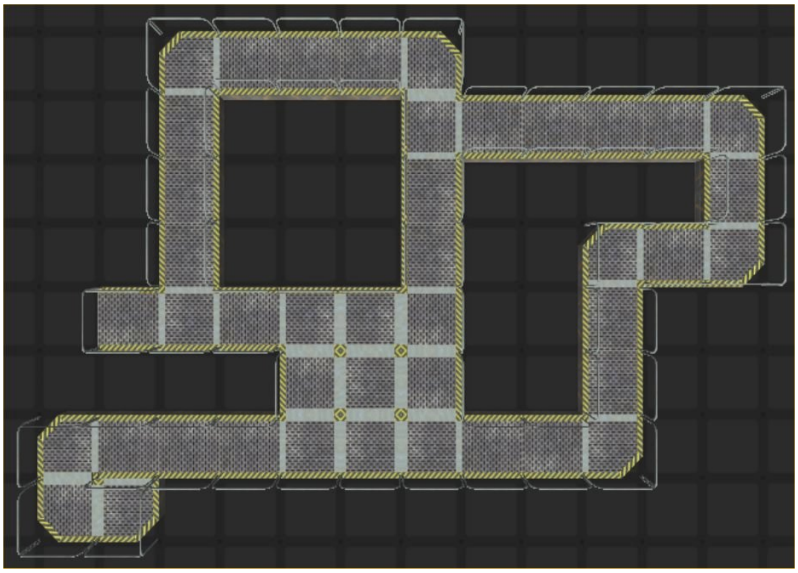
You will find gantries in the "platforms" section of a segment media library. Their purpose is to create walkways and crossovers in rooms, like this:

🗁 "FPSC X10 Files/Mapbank/Tutorials/Gantry_example.fpm"

Gantries overlay onto solid segments like rooms. There are different shapes of gantries to allow the creation of complex gantry walkways, they include:

T Junction	Corner	Cross over	Dead end	Straight
				

By mixing these and using the R key to rotate them, you can create complex structures:



6.4.2 Lifts

There are three sections to lifts, a lift base, lift tube and lift top. Whenever you build a lift you at least need a lift base and lift top. The height of the lift depends on how many lift tubes you put in between the base and top.

Base of lift	Tube section	Top of lift
Lift Base	Lift Tube	Lift Top

Here's an example of a lift built into a level:

The pad on which the player is moved up and down the lift is located in the segment with the '(platform)' text in the name. There are two segments which contain this pad, one that starts at the base section and one that starts at the top section. Also note that enemies cannot operate lifts or follow you into them.

When you build a lift shaft you also have to use doors to provide entrances and exits.

An example of two lifts can be studied in this demo map:

🗁 "FPSC X10 Files/Mapbank/Tutorials/Lift_example01.fpm"

6.4.3 Teleports

Another cool way to add complexity and puzzles into your games is to use Teleports. There are two teleport types:

Teleport IN Segment	Teleport OUT Segment
Teleporter IN	Teleporter OUT

Simply place the teleport IN segment where you want a player to be teleported from. The game engine will then reposition the player at the location of the teleport OUT segment.

By renaming teleport IN and teleport OUT segments you can create multiple teleports within a level.

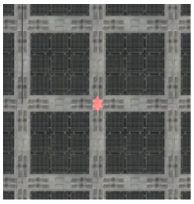
6.5 Waypoints

Neutral characters can move in many ways. FPS Creator X10 keeps everything open ended so you can decide how you want neutral characters to behave. One of the ways they can move is via waypoints. Waypoints are path routes that a neutral character will follow. They are easily created and a neutral character will automatically find and follow it when you set the Start condition of the character to 'No Action'".

By adding more nodes and with careful positioning of them it is easy to create complex path routes for the neutral characters in your game, e.g.:

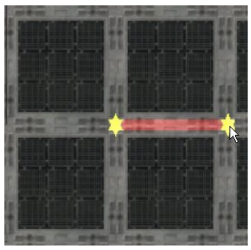
6.5.1 Creating a New Waypoint

When you click on this icon a waypoint node will be created in the centre of the screen, it is positioned at the ground level of the layer you are currently editing. The node is star shaped and looks like this:

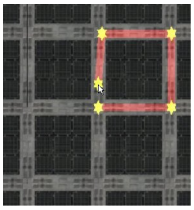


To add another node: Point the mouse at the node you wish to add another node to. Press the left mouse button while also holding down the shift key. A new node is created to the top right of the current node as seen here:

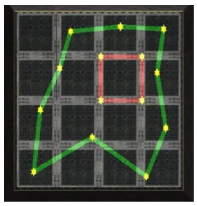
You can drag any waypoint node to a new location by left clicking on the node and whilst keeping the left mouse button held down move the mouse to its new location:



By adding more nodes and with careful positioning of them it is easy to create complex path routes for the enemies in your game, e.g.:



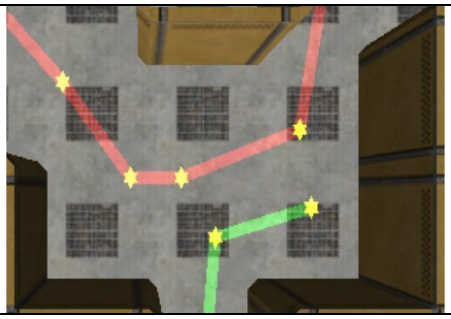
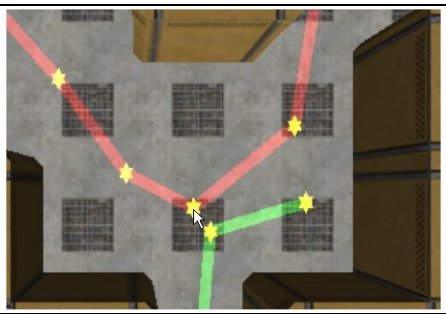
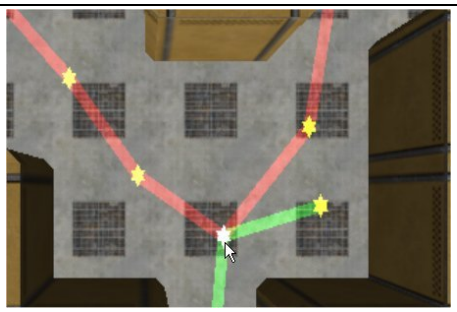
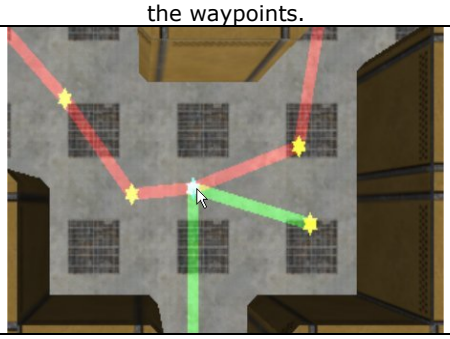
You can create more than one waypoint path. Click on the "create new waypoint" icon and start a new route. The new waypoint will be in a new colour to help you distinguish it:



6.5.2 Linking Waypoints

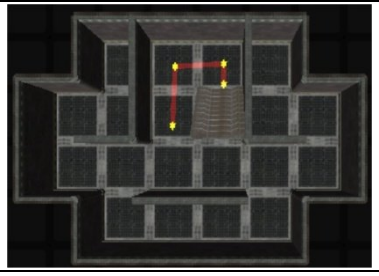
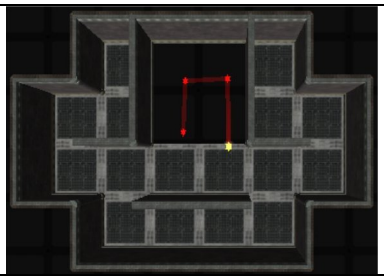
Way points can also link up with each other. By linking waypoints you can create multiple routes for neutral characters.

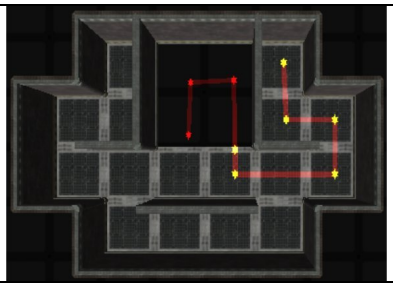
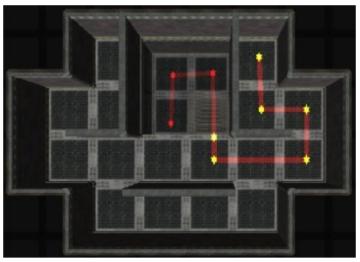
The process of linking is easy. Just create two or more waypoint paths and then drag a node from one waypoint on-top of a node from another waypoint.

	
Start with two or more way point routes.	Left click and drag one of the nodes from one of the waypoints.
	
Line the dragged node with the node on the other waypoint and release the mouse button. The nodes are now linked.	Click and drag the linked node and both waypoint paths will accommodate the movement.

6.5.3 Waypoints across Layers

It's possible to lay waypoints up and down the layers of your levels. This sequence shows how to achieve this:

	
Viewing Layer 5 with a waypoint ending at the	Left click on the end node and press + to move

foot of a staircase.	up to Layer 6. The lower layer waypoints are shown in red now.
	
Add more nodes to the path on Layer 6.	Tab pressed to reveal layer below.

🗁 "FPSC X10 Files/Files/Mapbank/Tutorials/Waypoint02.fpm"

6.5.4 Waypoint command list

Create new waypoint	Click on this icon
Waypoint editing mode	W key or click on this icon
Add new node to waypoint	Shift+Left Mouse Click pointing at a node
Delete a node from a waypoint	Shift+Right Mouse Click pointing at a node
Move a waypoint	Left Click and drag a node
Link waypoints	Left drag a node onto another waypoint node
Remove waypoint link	Right click on linked node
Show/Hide Waypoints	Press the space bar to show or hide waypoints. Use this when you are working with segments and entities and want to hide the waypoints that may be disrupting your view.

6.5.5 Waypoints for characters moving between layers

Waypoints are primarily used by neutral characters to define pre-designed walk paths, however waypoints can also be used by enemy and ally characters to get from a high place to a low place, and vice versa, but they will only be able to do this if the waypoints are designed in a specific way. Ensure that your waypoints start and end on floor that is unobstructed by static obstacles, and that the character has enough space to reach the centre of both start and end positions. Also ensure the waypoint only traverses a single layer at a time. If you want a waypoint to ascent five sets of staircases five layers high, you must create a waypoint for each layer, so the final waypoint would have six points from start to end.

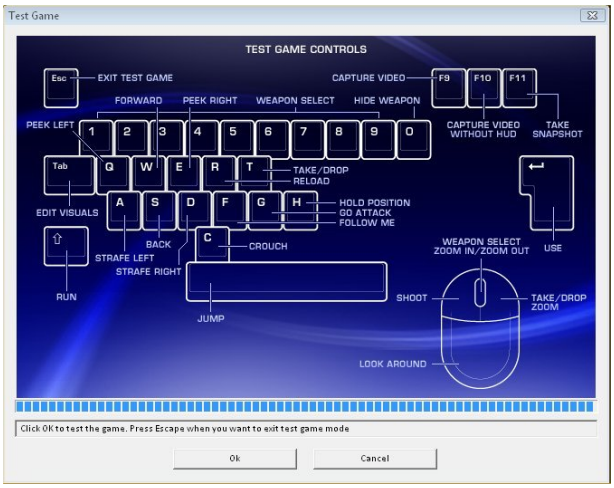
The other key use of waypoints is to allow neutral characters to walk a set path within the level, such as a guard on patrol or a robot conducting its daily tasks. To set up a character for this type of behaviour, you first need to lay down a series of waypoint nodes near the character, ensuring that at least one of the nodes is beneath the character. You then need to edit the properties of the character to set its alignment to NEUTRAL, and set all four reactive conditions (start, see, trigger and hurt) to NO ACTION. The character will then default to following the nearest waypoint node within their maximum range.

7 Testing and Previewing Levels

The job of the editor is to provide you with a visual representation of your game levels. The game play aspects can only be seen once you actually "Test Level" or "Build Game". When you are editing, you are essentially editing just one of the levels from your game. At any time you can try out your level by using one of two icons from the tool bar.

7.1 Test Level

When you click on this icon FPS Creator X10 creates your level from the segments and entities painted into it. The time it takes to construct the level will depend on the complexity of the level. Once the process is completed you will see this window, click the OK button to test the game.



This will launch you into the constructed level which you can try out from a game play point of view. To play the game, press W/A/S/D to move, Space bar to jump, C to crawl, hold Shift down to run, Enter key to Use objects like doors, Q and E to peek around corners (enemies will not see you). Weapon selection is on the 1 through to 9 number keys on the main keyboard. To reload a weapon press R and to fire press the left mouse button. Some weapons have a zoom lens, use the right mouse button to look through these. You can also pick up objects by pressing the right mouse button.

Additionally, you are able to record your test game session to video if you wish using the F9 and F10 keys. F9 records the game in progress complete with HUD displays, F10 will remove the HUD from the recording so you only get the game scene footage. Creating video footage is great for using in your final levels as cut-scenes when combined with the Story Zone marker. You can also take a snapshot of the game in progress with the F11 key. The video capture system is supported by multi-core technology, so the more CPUs you have, the bigger and higher quality the video capture will be. The settings to control how the video is captured can be found in the SETUP.INI file located here:

🗁 "FPSC X10 Files\setup.ini"

You can locate your captured video footage at "FPSC X10 Files\videobank\" and your snapshots in the root folder of "FPSC X10 Files\".

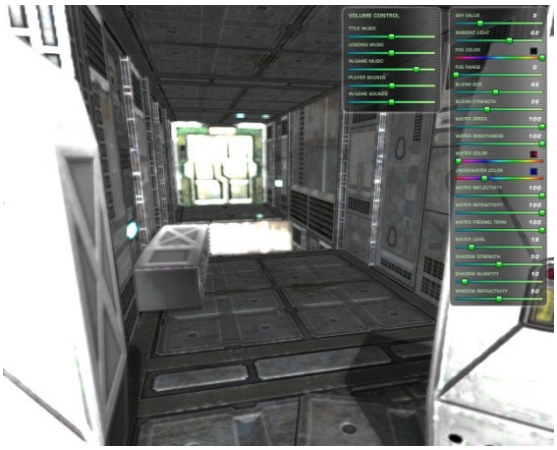
7.2 Effects Menu

During the TEST game process you can change your games' visual settings. Take this scene for example:

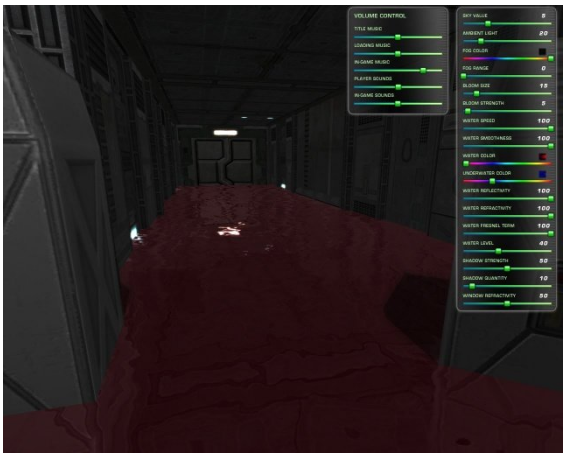
You might prefer to have this level of your game as dark and moody. Well it's a cinch to change this, just press the "Tab" key while you're in test game mode and you'll be presented with this control box:

By moving the "Ambient" slider to the left you affectively reduce the amount of ambient light from the game's scene, so you'll end up with something like this:

Maybe you want a brighter scene, just increase the ambience setting and also alter the "Bloom" setting for a result like this:



You also have control of the water from this control. By moving the level of the water and changing it's colour and how it ripples, you will soon have your own customised game level:



Fancy making a horror game? Why not add suspense to your level by using the fog option. By increasing the fog range you can make the player's view restricted from where enemies can attack from:

All the settings you change will be remembered. So as soon as you press Escape to come out of test game mode and back into the editor, FPS Creator X10 keeps a note of the changes you made and will save them when you next save your level.

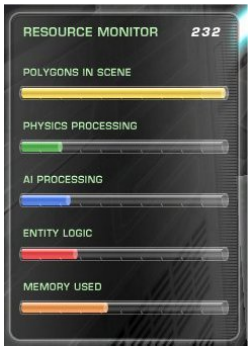
7.3 Volume Menu

Displayed to the left of the effects menu is the sound mixer:

Use it to perfect the music and sound levels for your game. It's great being able to do this while your game is playing.

7.4 Performance Monitor

Even Direct X10 will have its limits! Yes, you cannot just add and add enemies and expect the game to perform optimally. There are many events that FPS Creator X10 has to manage and the increase of entities adds work to the physics simulation and to the AI system. For this reason we have provided a performance monitor. Accessing it is easy, just press "Tab" again and it will appear:



The value in the top right is the frame rate of your game. If this drops then you need to optimise the section of the game you are playing.

Below the frame rate are five bars, these all give you some idea as to where the work is being done during the game.

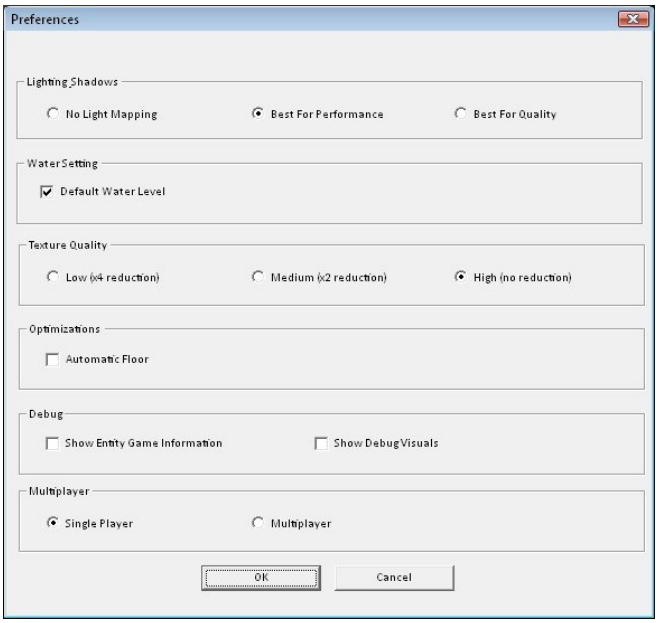
If the 'polygons in scene' bar is full, this means there is too much scene complexity which could be due to an intense shader task or a high number of polygons. Subdividing rooms, removing static entities and refractive elements like windows will help reduce this bar.

Let's say you enter a room where there are thirty enemies, as soon as these enemies see you they will start to attack. You may notice the 'AI processing' and 'Entity logic' levels increase, if they max out it would be advisable to optimise the room by reducing the number of enemies and dynamic entities.

If the 'physics processing' bar is full, this means too many dynamic entities or character ragdolls are in close contact. You can reduce this bar by lowering a characters Decay Time and not spawning as many entities in close proximity.

7.5 Test Game Preferences

The settings used in TEST game are taken from the Preferences dialogue:



The settings include Lighting and shadows, water settings, texture quality, optimisations, debug information and multiplayer option.

Lightmap Settings

A look of quality can be added to your game by using shadows. FPSC X10 uses a well know system called light mapping to achieve realistic shadowing. Lightmaps are extra details added to the geometry of your level and must be calculated when the game is built. Choose from one of three following settings: No Lightmapping, Best For Performance and Best For Quality. For more details on each see the extra information in the next Chapter.

Water Settings

By default the level is provided with a water plane that can be adjusted in the X10 effects menu, however you may choose not to have any water in your game at all. If this is the case, you can tick off the 'default water level' here to remove water from the game.

Texture Quality

Setting this to HIGH means the full size of the textures will be used, consuming the highest amount of graphics memory. If you want to reduce less graphics memory, you can choose Medium to divide the texture sizes by two and Low to divide the texture sizes by four. Lightmap textures are not reduced, and compression is not affected

Optimisations

You can choose to add an automatic floor with this flag. The floor style will be taken from the floor positioned at the very top left corner of layer zero.

Debug

Choose to view information on all entities that you encounter during the test game process. You can view additional visual information such as lighting as well.

Multiplayer
Choose the single player option to continue editing your level as a single player level. Select the multiplayer option to change the editing to an arena mode level. When you are in the arena editing mode, the editing grid will tint slightly red to indicate which mode you are in.

7.6 Quick Level Preview

If you just want to check how the geometry of the level is working you can click on the Quick Level Preview icon. This process will not add in the entities and it will skip things like light mapping calculations and thus allow you to preview the level far quicker.

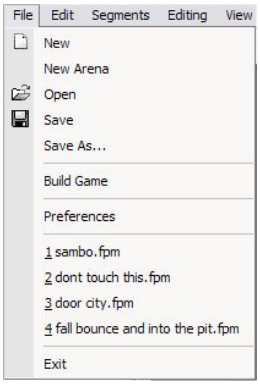
In preview mode there is limited collision detection. This can make it difficult to navigate the level. To get around this, the start position of the preview is based on the centre point of the editor. In other words, if you are viewing the centre of a room on layer 8 of your game and then press the quick preview icon, the camera will start at this point.

8 Building a Stand Alone Game

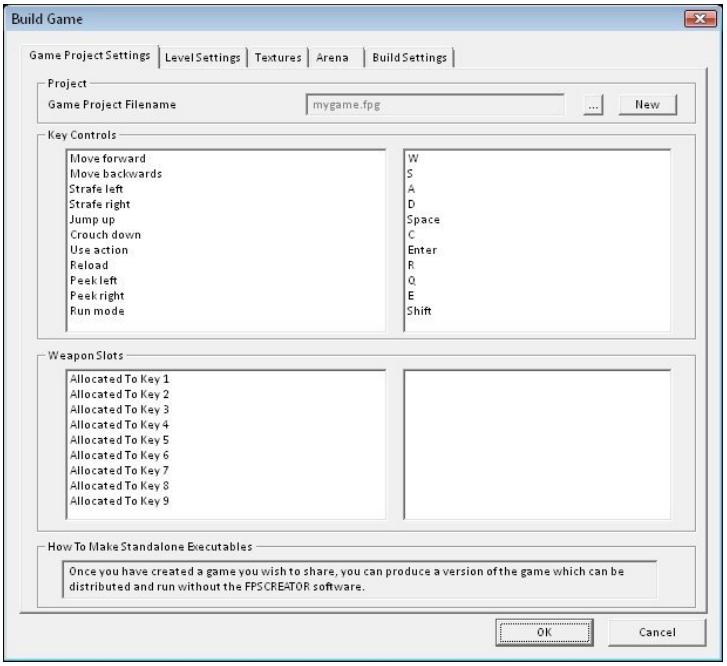
FPS Creator X10 provides a powerful "Build Game" feature that will take your maps and create a stand alone game file. This means you can pass your games onto friends and family for them to play.

It's important to understand that the build game system does not build the game from the level you may already have in the editor. It creates a game made up of one or more levels stored on your hard disk.

Build game is accessible from the File menu:



When you select Build Game you will be presented with the following window which is divided into four tabbed areas: Game Project Settings, Level Settings, Textures and Build Settings.



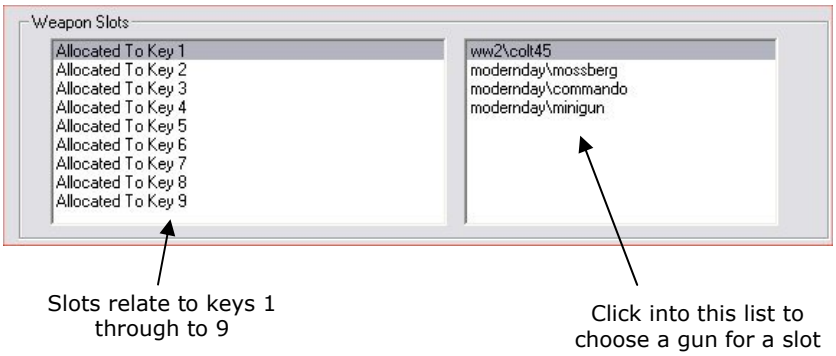
8.1 Game Project Settings

This is where you define three key things about your game.

Game Project Filename
This file holds information about how you want to build games each time. The default file is called "mygame.fpg". You can create a new project by clicking on the "New" button or you can select a previously edited file by clicking on the "..." button and browse for it. All changes you make throughout this build game process will be stored in the file selected here.

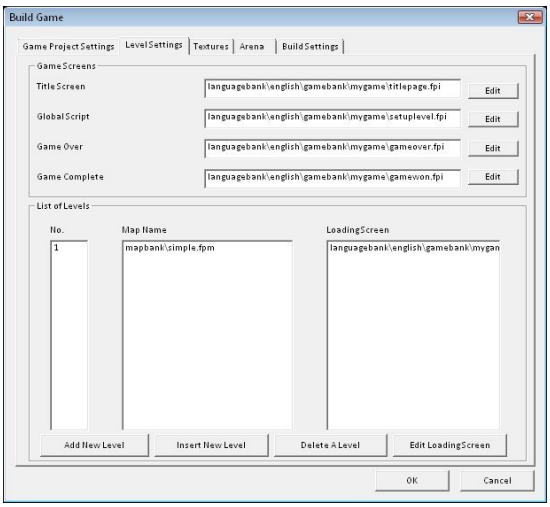
Key Controls
FPS games are very frantic at times, it's therefore best to keep the keys set to the defaults we have designed (as most game players are used to these). You can of course change and edit them should you wish to, and this is where you do it. Just click on the key window to change the key associated to the action and then press the new key you want to use for that action.

Weapon Slots
The player can have up to nine different weapons during a game. This list is where the user chooses for these weapons to be accessed from. Just click in the right window and a drop down menu will appear with the available weapons. Select a weapon and it will be assigned to that slot. It's important to note that the weapon must be placed into the level so that the player can pick it up during the game. This list does not give the player weapons, it simply describes where the weapon will be accessed from once it's been collected.



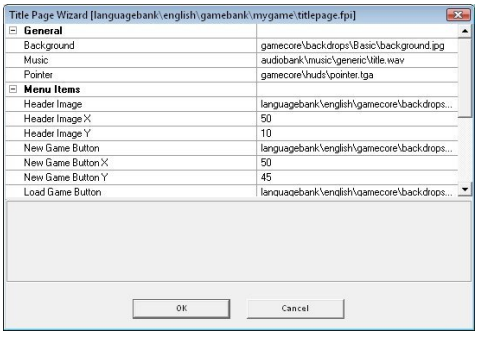
8.2 Level Settings

You can have a main menu in your game and multiple levels. You control and edit these aspects of your game from this section of the Build Game window.



Game Screens
The first section lists items that apply to the game menu screens. You can leave them as they are or edit them to give them your own personal touch. The screens are all controlled by fpi files (script files).

Title Screen
The title screen defaults to the script file "titlepage.fpi". You can EDIT this fpi file by clicking on the EDIT button next to the filename. Doing this will bring up this dialog box:



The Wizard is easy to use. If there is something within it that you want to change then just click into the right hand window and change the current setting. The first property in the "General" section is the background image that will be used for the title screen. Try to change it by clicking on the file text and replace it for another image from your hard drive.

Important: You must ensure that all extra media that you use in your games should be stored within the FPSC X10 Files folder layout.

Image formats supported are Jpeg (jpg) and dds. (Tip: a great tool for viewing all types of image formats is the free utility called Irfanview, download it from www.irfanview.com).

The properties you can alter for the title screen include:

Background image	This is the image used at the very start of your game for the background title page.
Music file	The music file to be played when the game's title page is displayed (must be 8 bit pcm wav format).
Mouse pointer used for selecting	The pointer image to select the menu items. (default is pointer.dds).
Menu header image and X, Y positions	This is the image used to display "GAME MENU" onto the title page. You could of course change it to the name of your game, eg "DEATH ZONE"! The X and Y values represent the position on the screen that this image will be positioned.
New Game image and X, Y positions	The image displays the "NEW GAME" menu button.
Exit Game image and X, Y positions	This image displays the "EXIT GAME" menu item, located at the X,Y position chosen.

Advanced Users: Note that this is just an editor for modifying the file titlepage.fpi. If you want, you could use Notepad to edit the fpi and make a more complex title page!

Global Script
The global script file controls how the game's pause menu works and the various visuals that make up the head up displays.

The Preview button allows you to see the changes you have made.

General
First there are some menu options for background, music and pointer.

Music	This is the music that will play during the game's level.
Pointer	The image of the pointer that will be used during the pause menu.

Level Setup

Sky	The 3D Sky box (.x file) used in the game. FPS Creator X10 is supplied with a selection of different sky boxes for you to choose from.

Screen Overlays

Fade Out Image	This image is used at the end of the level for fading from the game view to whatever this image is. The default is just a black screen. You could fade to a different colour like white or even some image representing a job well done.
Eye Damage Image	When the player gets hit by enemy bullets, this image is quickly flashed onto the game screen.
Default Zoom Image	If you create a weapon that has a right click zoom lens, then this image will be used if no other has been configured.

Status Panels

Lives Image, X and Y positions	The image that shows how many lives the player has is set here, along with the X and Y position where it is placed on screen, example:
Lives Numeric, X and Y positions	The style of font that the number will be displayed and the x,y location of where it will be displayed.
Health Image, X and Y positions	The image representing the health of the player. Example:
Health Numeric, X and Y positions	The text style used to show the values of the player's health.
Air Image, X and Y positions	The image of the HUD box holding the numeric for the player's remaining air.
Air Numeric, X and Y positions	The style of font used to show how much air is left to the player before damage is dealt.
Score Image, X and Y positions	The image of the HUD box holding the numeric for the player's current score (if the score goal value is set).
Score Numeric, X and Y positions	The style of font used to show how the current score the player has achieved so far in the game.
Ammo Image, X and Y positions	The image of the weapon used to fire the ammo. If a weapon does not provide this image of itself then the default weapon will display.
Ammo Numeric, X and Y positions	The style of font used to show how many rounds are in the current clip and the total number of rounds also held.

Menu Items

Header Image, X and Y positions	The header text used on the pause screen (after player has pressed Escape).
Load Game Button, X and Y positions	This button is clicked to bring up the load game dialog
Save Game Button, X and Y positions	This button is clicked to bring up the save game dialog
Saving Bar Image, X and Y positions	Image is used to inform that the game is being saved
Exit Game Button, X and Y positions	The EXIT GAME text used on the pause screen.
Continue Game Button, X and Y positions	The CONTINUE GAME text used on the pause screen.

Game Over

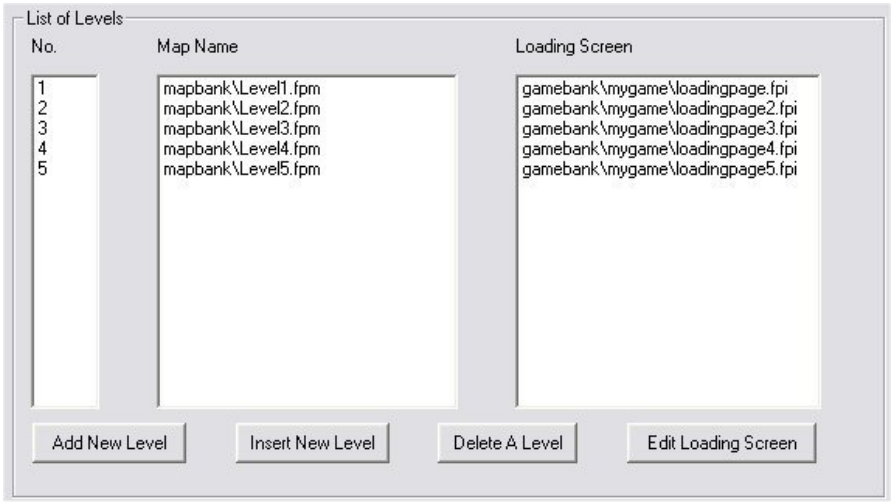
Background	The image shown when the player has lost all lives.
Music	The game over wav file to be played when this screen appears.
Header Image, X and Y positions	The text image that is pasted onto this game over screen.
Timeout	The time delay before the game reverts back to the title page.

Game Complete

Background	The screen to show when the player has completed the game.
Music	The triumphant music to be played when the game is finally done.
Header Image, X and Y positions	An image that can be pasted onto the screen, eg "GAME COMPLETED – WELL DONE!
Timeout	The time delay before the game reverts back to the title page.

List of Levels
The lower half of this tab provides the tools for setting the order of the levels in your game. A game can have up to 50 different levels.

Click on Insert New Level and choose a saved level from the hard drive. The selected level will then be listed in the centre column and it will be assigned a level number (shown in the left column) and a loading page fpi file (shown in the right column). Here's an example of a five level game setup:



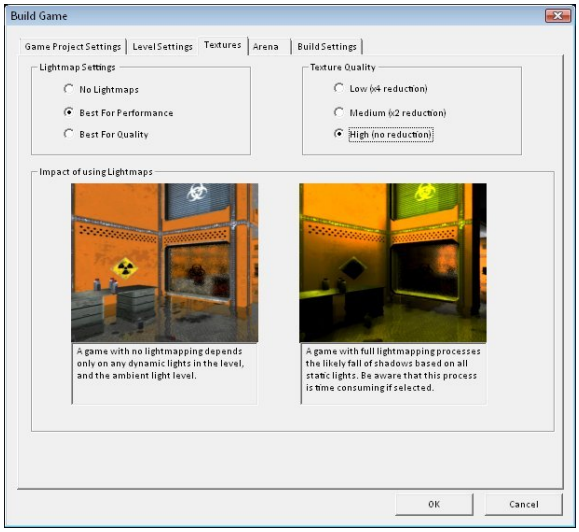
Using "Add New Level", "Insert New Level" and "Delete A Level" buttons you can organise the levels how you want. You can even repeat levels if you want!

FPS Creator X10 will ensure that all weapons and stats are copied across each level as users play your game. So if you pick up a weapon on level 1 you can expect to have it in your arsenal for all subsequent levels.

The Edit Loading Screen button allows you to edit the parameters used in the loading screen FPI file. This means you can customise the background, music, display and style of loading bar.

8.3 Textures

The textures tab controls the lighting and visual detail of your game.


Build Game
Game Project Settings
Level Settings
Textures
Arena
Build Settings
Lightmap Settings
No Lightmaps
Best For Performance
Best For Quality
Texture Quality
Low (x4 reduction)
Medium (x2 reduction)
High (no reduction)
Impact of using Lightmaps
A game with no lightmapping depends only on any dynamic lights in the level, and the ambient light level.
A game with full lightmapping processes the likely fall of shadows based on all static lights. Be aware that this process is time consuming if selected.
OK
Cancel

Lightmap Settings

Choose from one of three following settings:

No Lightmaps

Best to use this when you want a quick build of your game to test it out. You therefore cut out any lightmap calculation during the build process and get to play the game far sooner.

Best For Performance

Generating soft lightmaps is quicker for FPS Creator X10 to do. They are very realistic and will work in many areas of your games design. The downside is that the lightmapping does not account for transparent textures, and may cast shadows from parts of the scene that are rendered invisible due to transparency.

Best For Quality

Use this option to create the ultimate realism for your levels, which include the calculation of transparent textures. Massive calculations have to be computed so don't expect a fast result. It's best to calculate this when you know your game is finally completed game play wise, so you can leave FPSC working away on the maths. For large or complex games this process can take hours (PC specification depending) so use this sparingly, or perhaps leave it mapping overnight.

Texture Quality
Everything you see in your FPS games has some kind of texture describing what it looks like. Take for example a door, its texture looks like this:

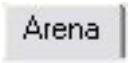
All textures in FPS Creator X10 have been created between 256 x 256 and 4096x4096 pixels. Texture sizes directly affect whether your final game will run on a graphics card with a minimal amount of video memory. FPS Creator X10 lets you select one of these options:

- Low = x 4 reduction
- Medium = x2 reduction
- High = no reduction

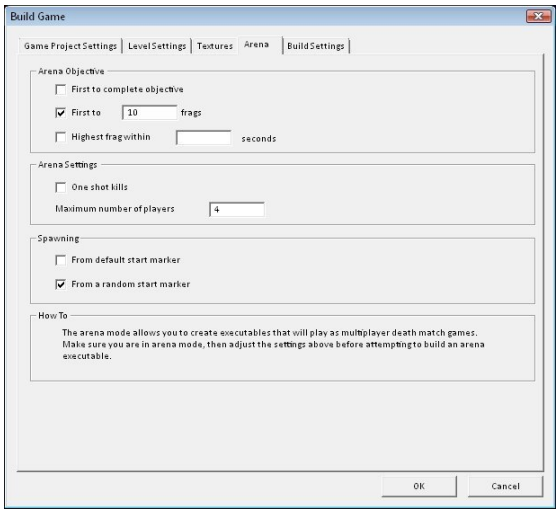
By using the options of low, medium and high you can set the texture quality used throughout the final game. The reason for doing this is that you may find that very complex levels that use many textures start to exceed the power and memory of some video cards.

Each entity within a game can also be tagged within its own properties to ignore this global setting. The property is called "Reduce Texture". For example you might want a global low setting but where there are signs in your game a player needs to read you want them to be high resolution.

8.4. Arena Executables



If your game is to be a multiplayer arena game, then this tab will be available to you for fine tuning how the arena game will work.



Arena Objectives

First To Complete Arena

If the level is setup with a win-zone or other objectives, the first player to complete all the objectives wins the arena.

First To X Frags

The first player to reach or exceed the number of frags entered here wins the arena.

Highest Frags In X Seconds

The player with the highest frags when the time expires wins the arena.

Arena Settings

One Shot Kills

Tick this box to activate the feature whereby any shot successfully made with any weapon instantly kills the target. Note that if you use a weapon with a zoom scope, you can kill any player with one shot regardless of this setting.

Maximum Number Of Players

Set this value between 2 and 8 to limit the number of players that can join the game. You may wish to have eight characters in the arena to choose from, but only allow four players at any one time. The more players you allow, the more work the server machine has to do and the more likely you are to have lag issues.

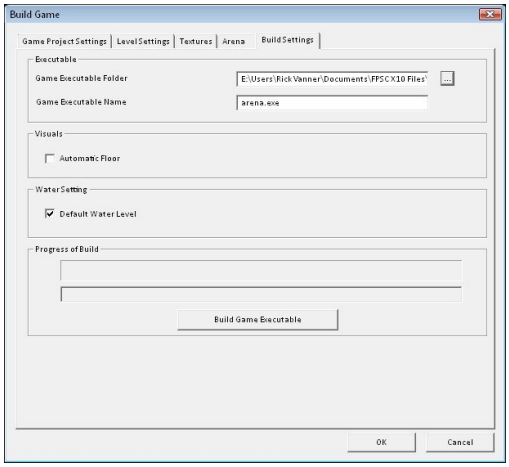
Arena Spawning

Random Spawn Position

Tick this box to allow the player to spawn from any other characters spawn position, instead of the player characters own spawn point.

8.5 Build Settings

Having set up the previous tabbed data pages you are now ready to start the build process. The Build Settings tab is divided into three key areas, Executable, Visuals and Progress of Build:



Executable

The "Game Executable Folder" shows you where your new stand alone game will be created. The default is set to "FPSC X10 Files\X10 Executables\".

The filename can be edited within the "Game Executable Name" text field.

Visuals

Use the 'Automatic Floor' flag when you wish to set your level in an outside location and require the benefits of a horizon. The actual editing area of the level is 40x40 tiles, which means your landscape will be cut short as your player encounters the edge of your map. By adding automatic floor, you can continue the appearance of a floor moving off into the infinite plane for better visuals, even though you cannot traverse the fake additional floor. The floor will be constructed at layer zero, the lowest point of your editing environment. The floor will default to a grey tile pattern for generic indication of the feature. To use a floor type of your own such as moonscape, grass or rock you simply place a single segment at layer zero, in the top left corner (scroll up and left, then use the [-] key to move to the lowest point in the depth of the editing environment). Even this automated floor is not visually convincing when viewed from a high location as the horizon cuts off at a distant camera clipping range, so it is recommended you design your levels strategically so views into the outside world are controlled for maximum visual effect. This flag is one option to help you achieve this effect.

Water Settings

Use the 'Default Water Level' flag when you wish to switch off the water plane in the game, so the game uses no water whatsoever.

Progress of Build

Finally you will be ready to build your game, just click on the Build Game Executable button and the process will begin. You will see a progress bar work across from left to right. When the game is built you will be returned to the Editor.

Using Windows Explorer, navigate to where your game was created and run the exe file.

9 Physics

9.1 How Physics Works

You have the ability to use real physics in your single player game (multiplayer arena games do not support physics). Basically, physics means that crates fall over realistically, bottles smash, tall structures fall, one object can knock over another, and so forth. You have the option of changing this behaviour based on the demands of your game level. Each entity has a number of properties that control exactly how they interact within the game world, including the entities weight, how much friction it applies against other entities and whether they explode.

9.2 Creating physics based situations

As the physics system is applied automatically by default, there is no long and drawn out method of producing physics within your level, you just drop in an entity you like and it instantly responds as it would in real life. There are some basic scenarios you can create however to demonstrate the various effects you can place in your games to impress those playing your game.

- Crates Do Fall – Select a crate from the entity library and place it on the floor. While the crate is still selected, hit RETURN to find the top of the crate you have just placed. Now click the left mouse button to place a crate on top of the first. Do so again so you have three crates stacked on top of each other. Now click TEST GAME and find the stack you have made. Run at the stack and jump as you reach it, notice how the player knocks the crates over.

- Stairs Are High – Select a cylinder from the entity library and place it at the top of some stairs you have previously created. Now run the TEST GAME and run at the cylinder so that you push it down the stairs. Notice how it bounces off the scenery as it goes.

- Duck Shoot – Create a room with a long table at the far end, then go to the entity library and find a bottle entity. Select it and place it on the long table. Right click the bottle and change the properties so that the bottle has only 4 strength. Now left click the bottle to pick it up, then place copies of the newly changed bottle along the table to give you something to shoot at. Right click to end the bottle placement. Go to the MARKERS tab and select a Player Start marker down, right click it to edit its properties, then change the HAS WEAPON field to a pistol. Now TEST GAME and shoot the bottles to see them explode. You can further modify the scene back in the editor by switching the EXPLODING field to zero, causing the bottles to smash without the explosion. Make sure the distance between you and the bottles is great when the bottles are set to explode.

9.3 Tricks and tips when setting up physics

- Typically keep the physics weight and friction settings as provided by default for each entity. They have been manually set to represent the real behaviour of the entity. Changing them may cause them to behave in an uncommon manner.
- Switching physics off for an entity will cause that entity to cease reacting to the game world, and it will even ignore gravity and other entities. It will however move if its AI uses waypoint movement or other motion actions. Useful for a faster variety of entity.
- Do not populate a room with too many physics entities, and try to use them cleverly to aid the story and enrich the gameplay. Adding a hundred bottles on a shelf that plays no key role in the game will just slow down the performance and make the local areas around those bottles run slower. Remember within the game engine, every physics object must check itself against every other physics object at some level. The more you use, the slower things become.

10 Multiplayer Death Match Games

10.1. How Multiplayer Works

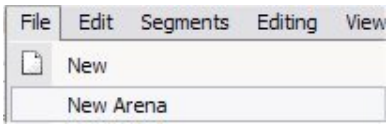
You have the ability to create multiplayer death match games, and play them online with your friends. The creation method is the same as a single player level, except for some exclusions, and you simply have to select multiplayer mode and save out the executable as normal. DirectPlay is used to control the multiplayer experience, and uses the server-client model for network communication. You can run your arena games on a local network, or over the internet using the UDP protocol. The host player (server) must have a valid IP address, not an internal network IP (such as you would find if your computer was sharing an internet connection). Joiners do not need to know their IP, they simply need the exact same executable as the hoster and also need to know the name of the player hosting the session. You CAN use an internal network IP if ALL players are on the same local network (LAN), but this is no good for Internet multiplayer. Also remember that your computer may not be the main computer connected to the internet and so will not be able to host indirectly. You will almost certainly have a firewall in place as well, and you must ensure the final executable has permission to communicate through your firewall and that the appropriate ports have been allocated to allow the host and joiners to connect. The arena games use UDP port 2303 for inbound and outbound communication.

10.2. Creating a Multiplayer game

Switching to and from Arena Mode

In order to create a multiplayer arena level, you must be in Arena Mode. There are two ways you can get into Arena Mode:

- Select the NEW ARENA from the FILE MENU



- Select PREFERENCES from the FILE MENU and tick the Multiplayer box

Using either of these will set the editor into the Arena Mode and block off certain areas of the library and property panels. You can toggle between arena mode and single player mode, in order to carry your level designs between the types of game you would like to make, but features that may be active in single player will cease to function when run in arena mode or built into an arena executable.

Aside from the differences mentioned in the next section, the editing of arena levels is identical to the creation of single player levels. When you are ready to build your arena executable to share and play online, you must do the following:

- Select BUILD GAME from the FILE MENU
- The BUILD GAME will always be in the mode you are editing, so if you were using a single player project file, you will now find the default arena level

project file in the Settings tab. When you create a new arena, a default arena project file will be created for you.
- Choose the game building choices as you would in single player mode, and then click the Arena tab which will have been made available once you are in Arena Mode.
- From here, select your Objectives, Settings and Spawn choices
- Go to the final tab, and enter a name for your arena executable
- Then click the BUILD EXECUTABLE button to build your game

You may now wish to share your game with your friends so you can play online. When the build has completed, and you have been returned to the editor, exit the editor software and locate the X10 Executables folder. You will find your newly created arena executable in its own folder. You may zip up the entire folder, and it is always a good idea to rename the zip-file with the day and time the executable was made.

You can send your newly created zip-file to your friends in many different ways depending on the size of the file and your capability to exchange files over the internet. Here are some suggestions:

- Use MSN Messenger to send the file to all the friends you wish to play against
- If the file is smaller than 20mb, you may be able to email the game to your friends
- If you have access to an FTP site, you can upload the file there and provide the link
- You can burn the zip file to a CD or writable media of sufficient size and deliver it to those people you wish to play against
- Some online email providers also provide a small amount of shared space, into which you could place the file and give your friends access to the area
- With the popular use of P2P file exchange software, you can create a folder on your machine that would commonly store your arena games, from which your friends could gain access and obtain the game

10.3. Differences between Arena and Single Player

There are some key differences between regular single player level and arena level editing. The most apparent is the lack of physics for the entities, and no dynamic entities that are not essential to the gameplay. These restrictions will be placed automatically in the editor when you are in arena mode to avoid confusion. The reason these restrictions exist, is due to the lag time that exists during multiplayer games. Game data has to be sent between all the PCs playing the game many times per second, so it's vital to keep this data as small as possible. Here is a list of restrictions:

- No physics on entities
- No dynamic entities except weapons, ammo and health
- All dynamic entities not essential to gameplay are forced to static
- No projectile weaponry (rocket launchers, grenades)
- Only one level per arena executable

11 Guidelines

Here are some key points to remember for creating good levels:

Level Editor Guidelines

- Do not create extremely large open areas or rooms as they will require a large number of polygons to render, which means your game will be slow in places
- Create a linear flow through the level and give the player clues where they should go next, the easiest level to design is a succession of rooms and corridors leading from the beginning to the end.
- Separate your rooms with corridors or narrow turnings, as this will allow the game engine to skip drawing large parts of the scene as it will not try to draw around corners. This will speed up your game's performance.
- Do not place too many dynamic entities into a confined area such as a high concentration of characters, items or weapons. Dynamic entities are rendered individually and the more you have, the more work the engine has to do and the slower your game is.
- Where possible, use static entities to populate rooms and areas. Static entities are added to the final scene geometry and are rendered together, meaning the game will draw the scene faster and performance will improve.
- On rare occasions, some segments do not work well together such as a grass tile and a corridor. Some corridors are very wide and the grass tile may actually poke through the corridor and cause Z-clashing (flickering polygons) where they both try to share the same space in the scene. The solution is to keep grass away from corridors and design your level so that you cannot go outside and see the corridor alongside grass.

Level Design Guidelines

- Give the player enough weapons and ammo to defeat all the enemies in the level, and build in a buffer to allow players not as proficient to complete the level eventually.
- Always seal your rooms and areas so the game engine is only rendering what it absolutely needs to render. Rooms should have ceilings added to them, and all open areas should be considered carefully to ensure each direction has a cut-off. Two buildings on opposite sides of the map separated by grass is a very expensive render and will slow down your performance.
- Make use of corridors as they are pre-sealed and are good at keeping your rendered polygons to a minimum. When connecting corridors to rooms, remember to use the dead end segment so that the corridor connects gracefully with the room. Make sure the room has a wall facing the corridor piece, and cut through that wall with a door of your choice.

Game Play Guidelines

- Remember that dead enemies can be set to drop their weapons upon death which the player can then collect – this is a good way of making the player work for their guns!
- Step your levels in difficulty and introduce elements of your game one at a time so they can learn at a reasonable pace. Throwing everything at them in one go will discourage the player from continuing to play the game.

- Keep complex characters to three or four per room, and do not let those characters wander too far from their origin. Complex characters consume processing power and will slow down your game if too many are used.

Multiplayer Guidelines

- A good multiplayer arena design is based on the concept of high speed game play, not story telling or puzzle solving. Arenas should be compact, non-linear, easy to navigate and ideally no more than 30 seconds to any position from any other position.
- Remember that when building and distributing an arena executable, not all computers are of equal specification. Design your arenas so they use as few graphics as possible, and remember to use the maximum players limiter to help guide your users into a fast game play experience.
- An important element of a good arena is balance. Make your players starting health, weapon power, weapon choices, health and ammo pickup distribution such that the player does not die too quickly, or lasts forever. This means extensive testing against online players.
- Choose graphics that use low polygon counts to speed up the game action, and make clever use of L-shaped corridors to block off large expanses of scenery. Rendering a highly complex arena will slow down the action and ruin a good high speed shooting match.
- Limit your use of non-essential entities in an arena design. They may look good at first, but those extra CPU cycles could be better spent dedicated to game speed.
  - When playing an arena game with your friends, choose the machine that is the fastest for the HOST player. The host player acts as the server and actually runs all the logic of the game itself.
  - The arena executable uses a central listing service located at www.fpscreator.com to store all hosters and their IP addresses, so that the joiners can browse those games and join using a name instead of a complex IP address. If you are joining and cannot see any names in the list, you are either using 'different' executables (each executable has a unique code assigned to it), or the central server is down.
  - When editing in arena mode (as opposed to single player mode), some items will no longer be available to you such as the physics fields, non-essential dynamic entities, some weapons and items. You will also be limited to adding non-essential entities as static entities only. These measures ensure your online game remains fast, which is a key ingredient to online play.

Migration from XP to X10 Guidelines

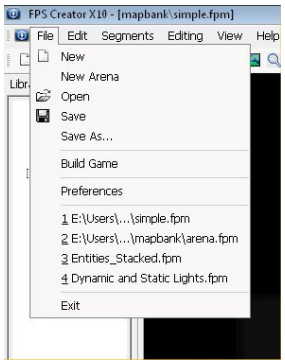
Users who are moving from the previous version of FPS Creator for Windows XP to the new FPS Creator X10 for Vista will be able to load their old levels and with some small changes be able to continue their projects to function in Vista and DirectX 10. There are a few migration issues to bear in mind before moving your old levels across:

- Always remember to make a back-up of your levels, so that you can continue with FPS Creator XP if for any reason you cannot migrate to X10.
- Once you have saved a level in X10, it can no longer be loaded into the previous version of the software.
- All characters in old levels will be using legacy FPI scripts which X10 no longer uses for AI behaviour. Change your main FPI script to DARKAI.FPI
- Some media has been renamed/removed from the new default library. You can copy missing media from your old FPS Creator, but do so with care!
- All shaders in X10 are Model 4.0. Your own shaders must also be SM4.0.

12 GUI Layout

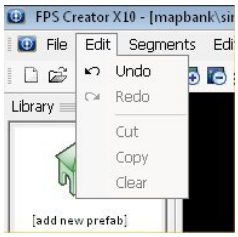
12.1 The Drop Down Menus

File



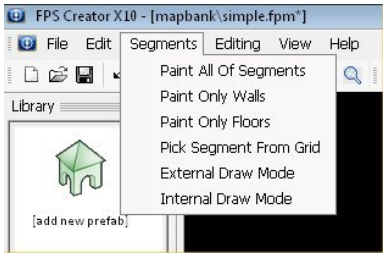
New...	This clears the editor window and allows you to start a new level from scratch.
New Arena…	Clears the editor window and sets you in Arena building mode.
Open...	This opens the file browser, so that you can select a saved level project. Projects have the extension ".fpm".
Save	The current project is saved when this is selected; if the project has not already been named then the "Save As…" dialogue will appear.
Save As…	The file browser is shown, enabling you to choose a filename and folder for the current project. If you choose a filename that already exists then you will be asked if you want to overwrite the old file.
Build Game...	This opens up the Build Game dialogue. Use this to build a final exe of your game.
Preferences...	A dialogue will appear into which you can select the preferences you want when editing a level and testing it from the editor.
1,2,3,4	The last four projects that were opened or saved in FPS Creator X10 are shown here; clicking on one of these opens the project and loads it into the editor.
Exit	This closes FPS Creator X10.

Edit



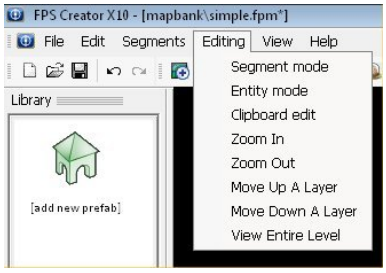
Undo	Undo the last action performed in the editor window.
Redo	Reverse the last "Undo" action.
Cut	Cuts the highlighted area in the editor window.
Copy	This copies the highlighted area in the editor window.
Clear	This clears the highlighted area in the editor window.

Segments



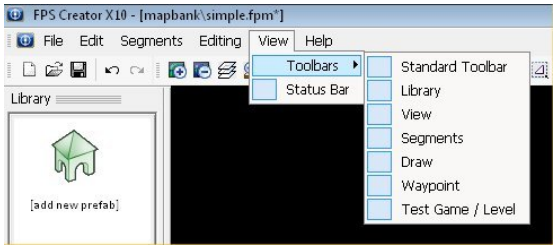
Paint All Of Segments	This sets the 3D cursor into a mode that ensures the whole of a chosen segment (floor and all walls) will be painted into a grid cell.
Paint Only Walls	This sets the 3D cursor into a mode that will only paint the wall sections of a chosen segment. The wall is painted in the direction indicated by the arrow on the 3D cursor. Use the "R" key to rotate the arrow.
Paint Only Floors	Ensures that only the floor section of a chosen segment will be painted at the 3D cursor position.
Pick Segment From Grid	Sets the cursor into "pick" mode. You must then choose a grid position that has the segment you want to use. Once chosen, you can start painting with it.
External Draw Mode	Will set the editor into External draw mode.
Internal Draw Mode	Sets the editor into Internal draw mode.

Editing



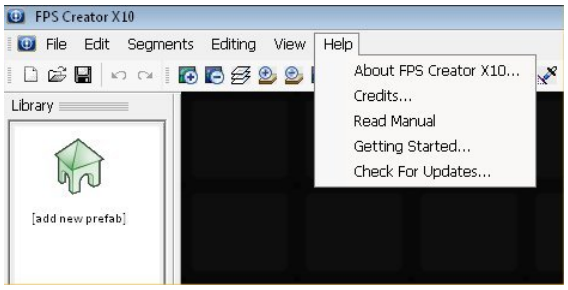
Segment Mode	Sets the editor into Segment editing mode.
Entity Mode	Sets the editor into Entity mode.
Clipboard Edit	Sets the editor into Clipboard mode.
Zoom In	Zooms in towards the 3D scene in the editor.
Zoom Out	Zooms out from the 3D scene in the editor.
Move Up A Layer	Moves the editing view up one layer.
Move Down A Layer	Moves the editing view down one layer.
View Entire Level	Shows a zoomed out view of the level. Simply click into the view to zoom to the area you want to continue editing.

View



Toolbars	The toolbars can be enabled and disabled here. When the sub-menu appears, just toggle the sections of the tool bar you want shown or hidden.
Status Bar	This enables/disables the status bar at the bottom of the screen.

Help



About FPS Creator X10…	A window appears that shows the version number that you are using, along with a link to the official FPSC X10 website.
Credits...	Shows a list of the team who created FPS Creator X10.
Read Manual	This calls up the PDF user guide for FPS Creator X10. You will need Adobe Acrobat to view a PDF file. See www.adobe.com
Getting Started	Calls up the Getting Started dialog box. From here you can access video tutorials and example maps.
Check For Updates	Calls up the Auto Update dialog box, which can be used to check the main FPS Creator X10 server to see if there are any new updates available for the software. The downloading and installation of the updates are automated to make the update process as easy as possible.

12.2 Toolbars

Standard Toolbar

	New	Clears the editor window.
	Open	Opens the file browser so that you can select a previously saved project.
	Save	The current project is saved when this is selected; if the project has not already been named then the "Save As…" dialogue will appear.
	Undo	Undoes the last segment edit action.
	Redo	Redoes the last undo action.

View

	Zoom In	Use this to get a closer view of the map.
	Zoom Out	Use this to get a wider, less detailed view of the map.
	Toggle Layers	Switches between single and multiple layer views. It has three modes, current layer with lower 5 and higher 5 layers, current layer only and half of current layer.
	Move Up Layer	Move up to a higher layer on the map editor.
	Move Down Layer	Move to a lower layer on the map editor.
	Entire Level	Zooms out to show the entire map area. Click into the map to view that area.
	Close Up View	Zooms in to the centre of the editor window, showing a rendered view of the contents of the middle of the screen. Right drag and move the mouse wheel to view around and left click to exit this mode.

Objects

	Entity edit mode	Sets the editor into entity edit mode
	Segment edit mode	Sets the editor into segment edit mode
	Interior	Chooses internal segment editing mode.
	Exterior	Chooses external segment editing mode.
	Wall	After clicking this, only walls of segments will be drawn.
	Floor	Sets the segment painting mode to only draw the floor section. Ideal for adding a ceiling onto the top of a room without affecting other walls.
	Pick Segment	Allows you to choose a segment type from the main editor window. It can then be drawn to any other part of the map.
	Select Area	Allows the user to highlight an area within the main editor window; this can then be erased, copied, moved, etc.

Draw

	Draw Segment As Line	A free drawing tool; this can be used to draw rooms and corridors in any shape or size.
	Draw Segment As Rectangle	Draw a segment in a square or rectangular shape. This is a quick way to set out large rooms.
	Draw As Ellipse	Draw circular rooms.
	Spray Segment	Spray the chosen segment over the map area.
	Increase Draw Brush Size	The segments are drawn over a wider area.
	Decrease Draw Brush Size	The segments are drawn over a narrower area.

Waypoints

	New Waypoint	Add a waypoint to the map and also enter into waypoint editing mode.
	Select Waypoint	Enter waypoint editing mode. You will need to use this when you have been placing segments or entities down and want to re-edit existing waypoints.

Test Game/Level

	Test Level	Test the current Level using the settings chosen in the Preferences dialogue box.
	Quick Level Preview	Test the current level geometry with just basic collision. No light mapping or entity AI will be setup. This is just a quick way to try out your level's design.

13 How-To Guides for Advanced Users

13.1 Game Resolution

The games made by FPS Creator X10 use the resolution of the Windows desktop, and as such has the benefit of being able to seamlessly toggle between the game and other applications. There is no upper limit to the resolution you can use as your game will scale the HUD graphics and layout accordingly, including the adjustment for aspect ratio.

13.2 Making New FPI LOGIC

You can create your own AI scripts using the FPI script language. FPI files control everything in FPS Creator X10 from the title page appearing to how your entities behave in the game. There are places within the interface which accept the use of an FPI file such as the AI fields of the entity properties or the screen descriptions in the Build Game options. FPS Creator X10 does not have its own tool to create these FPI files but because they are plain text files you can use your own favourite text editor, or Notepad which is provided with all versions of Windows, to create or edit FPI files.

Creating a Simple FPI File

1. Open Windows Explorer and navigate to FPSC X10 Files\scriptbank
2. Create a new folder called MyScripts\
3. Inside this folder, right click and select New->Text Document
4. Rename the newly created file to mylogic.fpi
5. Double click this file and select for it to be opened with Notepad
6. Type the following

```
desc = Loop Sound When FPI Used
:state=0:state=1,sound=audiobank\atmos\alarmhum.wav
```

7. Notice the colon (:) at the start of the line and after the first expression
8. STATE=0 is the condition, which means 'if the variable STATE is zero'
9. SOUND="file" is the first action which tells the game to play this sound
10. STATE=1 is the second action which sets the STATE variable to 1
11. The second colon (:) separates the conditions from the actions
12. The comma (,) separate individual conditions and individual actions
13. You only have one variable per FPI file, which is called STATE
14. LOOPSOUND will instruct the game to repeatedly play the WAV file

Using an FPI File

1. Run FPS Creator X10.
2. Paint a piece of floor in the map, then add any dynamic entity to it
3. Right click on the entity, and select the file browser button to the right of the 'AI Scripts/Main' field (this is the field which controls the main logic for this entity)
4. Locate and select the mylogic.fpi file you have previously created
5. Left click on Apply Changes and then click TEST GAME
6. When the game starts, the WAV file will be played as instructed

How FPI works

The FPI files are scanned in at the start of the game and converted to a fast form of byte code. These FPI scripts are then linked to the entities that use them and for as long as the entities exist within the game they will run the logic they

contain. Essentially whatever you want your entity to think about or do can be performed from inside these scripts. Scripts control everything including items appearing, disappearing, opening doors, unlocking doors, collecting items, collecting weapons, destroying items and anything else that moves within the logic of the game. Scripts do not control character behaviour in FPS Creator X10, a function now performed using the DarkAI component built into the software. It is also worth noting that some of the FPI actions and conditions are redundant due to superseded functionality in the software and remain in the language for legacy support only.

FPI Language

This is a typical event action line. You can have many actions/conditions:

```
:condition[,condition][,condition]:action[,action][,action]
```

This is a comment:

```
; myscript comment
```

This is a condition:

```
Conditionword=conditionvalue1 [conditionvalue2]
```

This is an action:

```
Actionword=actionvalue
```

Conditions list

These are the condition words, which will perform actions if all are true:

NEVER	is never true
ALWAYS	is always true
STATE=X	is true when the value stored in the FPI script is equal to X
RANDOM=X	is true when a random value between 0 and X is equal to one
HEALTH=X	is true when health equals X
HEALTHLESS=X	is true when the health is less than X
QUANTITY=X	is true when quantity is equal to X
SPEED=X	is true when speed is equal to X
RATEOFFIRE=X	uses the RATE OF FIRE value from the entity properties to control whether this condition is true, zero being always, and 100 being never.
PLRHEALTHLESS=X	is true when the players health falls below X
PLRHEALTHGREATER=X	Is true when the players health is greater than X
PLAYERASSOCIATED=X	is true when entity has been associated with player (lift)
PLRDISTWITHIN=X	is true when player is within X units
PLRDISTFURTHER=X	is true when player is further than X units
PLRALIVE=X	is true when player is alive and X is one
PLRHIGHER=X	is true when player is X units higher than entity
PLRELEVWITHIN=X	is true when player can be seen within X degrees vertical
PLRELEVFURTHER=X	is true when player cannot be seen X degrees vertical
ANYWITHIN=X	is true when any other entity moves within X quarter tiles
ANYFURTHER=X	is true when neither player or entities are within the specified range X from this entity
PLRCANBESEEN	is true when player can be seen
PLRCANNOTBESEEN	is true when player cannot be seen
PLRHASKEY=X	is true when player has pressed the key denoted by the value X
PLRUSINGACTION=X	is true when player performs the USE action
SHOTDAMAGE=X	is true when damage taken exceeds the value X

IFWEAPON=X	is true when the weapon being used by entity is ready and X is one
CANTAKE	Is true if the player can take this entity
ACTIVATED=X	is true when the activation value of the entity equals X
PLRWITHINZONE	is true when player is within the trigger zone
ENTITYWITHINZONE	is true when an entity is within the trigger zone
PLRINGUNSIGHT=X	is true when an entity has the player is gun sights
NEARACTIVATABLE=X	is true when entity is being near activated
NEWWEAPONCANBESEEN=X	is true when the entity can see a weapon
NOISEHEARD=X	is true when the entity hears a broadcast noise from scene
ANYWITHINZONE	is true when any entity is within its zone
ANYKEYWITHINZONE	is true if any key entity enters the zone
RAYCAST=X Y	is true when the raycast hits something in front from X to Y units
RAYCASTUP=X Y	is true when the raycast hits something above from X to Y units
RAYCASTBACK=X Y	is true when the raycast hits something back from X to Y units
NORAYCASTUP=X Y	is true when no collision upwards X to Y
FRAMEATEND=X	is true when animation X is at an end
FRAMEATSTART=X	is true when animation X is at the beginning
FRAMEWITHIN=X Y	is true when animation X is within frame Y
FRAMEBEYOND=X Y	is true when animation X is beyond frame Y
ANIMATIONOVER=X	is true when animation X is complete
ALPHAFADEEQUAL=X	is true when the alpha value equals X
REACHTARGET=X	is true when the entity has reached its target
LOSETARGET=X	is true when the entity has got stuck after X attempts
HEADANGLEGREATER=X	is true when the angle of the head is greater than X
HEADANGLELESS=X	is true when the angle of the head is less than X
WAYPOINTSTATE=X	is true when the waypoint state value equals X. State 0 means the entity has not yet started following waypoints State 1 means the entity is looking for the nearest waypoint marker to start from State 2 means the entity is following a waypoint line to its current waypoint marker State 3 means the entity has reached the waypoint marker and needs to decide what to do State 4 means the entity has reached the waypoint marker and it splits off in more than one other direction State 5 means the entity has reached the very end of the current waypoint structure and requires a decision to be made. A state of 999 means the entity has been placed in zero-waypoint mode where the entity simply ignores waypoints.
IFMARKER=X	is true when there is a marker previously dropped by entity in scene
IFPLRTRAIL=X	is true when there is a trail left by the player in existence
ISDARKAISEEING	is true if a character using DarkAI can see the player
ISDARKAIHURT	Is true if a character using DarkAI is hurt below the hurt value
ISDARKAIFIRING	is true if a character using DarkAI is currently able to fire at the player, as instructed by its internal behaviour system
HUDSELECTIONMADE=X	is true when the user has clicked HUD button X
TIMERGREATER=X	is true when the internal FPI timer exceeds X in milliseconds
ESCAPEKEYPRESSED=X	is true when the Escape Key has been pressed
SCANCODEKEYPRESSED=X	Is true when the key indicated by scancode value X is pressed
HUDEDITDONE	is true when 'editable' HUD item X is used
HUDHAVENAME	is true when no player-name has been entered
EFFECTAMBIENCELESS=X	is true if the effect value is less than X
EFFECTAMBIENCEGREATER=X	is true if the effect value is greater than X
EFFECTBLOOMSIZELESS=X	is true if the effect value is less than X
EFFECTBLOOMSIZEGREATER=X	is true if the effect value is greater than X
EFFECTBLOOMSTRLESS=X	is true if the effect value is less than X

EFFECTBLOOMSTRGREATER=X	is true if the effect value is greater than X
EFFECTWATERSPEEDLESS=X	is true if the effect value is less than X
EFFECTWATERSPEEDGREATER=X	is true if the effect value is greater than X
EFFECTWATERSMOOTHLESS=X	is true if the effect value is less than X
EFFECTWATERSMOOTHGREATER=X	is true if the effect value is greater than X
EFFECTWATERREFLECTIONLESS=X	is true if the effect value is less than X
EFFECTWATERREFLECTIONGREATER=X	is true if the effect value is greater than X
EFFECTWATERREFRACTIONLESS=X	is true if the effect value is less than X
EFFECTWATERREFRACTIONGREATER=X	is true if the effect value is greater than X
EFFECTWATERFRESNELLESS=X	is true if the effect value is less than X
EFFECTWATERFRESNELGREATER=X	is true if the effect value is greater than X
EFFECTWATERLEVELLESS=X	is true if the effect value is less than X
EFFECTWATERLEVELGREATER=X	is true if the effect value is greater than X
EFFECTSHADOWSTRLESS=X	is true if the effect value is less than X
EFFECTSHADOWSTRGREATER=X	is true if the effect value is greater than X
EFFECTSHADOWQTYLESS=X	is true if the effect value is less than X
EFFECTSHADOWQTYGREATER=X	is true if the effect value is greater than X
EFFECTWINDOWREFRACTIONLESS=X	is true if the effect value is less than X
EFFECTWINDOWREFRACTIONGREATER=X	is true if the effect value is greater than X
VOLUMEINGAMEMUSICLESS=X	is true if the volume is less than X
VOLUMEINGAMEMUSICGREATER=X	is true if the volume is greater than X
VOLUMEPLAYERSOUNDSLESS=X	is true if the volume is less than X
VOLUMEPLAYERSOUNDSGREATER=X	is true if the volume is greater than X
VOLUMEINGAMESOUNDSLESS=X	is true if the volume is less than X
VOLUMEINGAMESOUNDSGREATER=X	is true if the volume is greater than X

Actions list

NONE	no action
DESTROY	destroy entity
SUSPEND	disable the entity permanently, and keep it visible
RAGDOLL	Turns the entity into a ragdoll if it is a character entity
FLOORLOGIC=X	if X is one, entity never leaves its Y position
NOGRAVITY	Switches off gravity for physics entity
RUNFPIDEFAULT=X	run a default FPI script by value X (0-init,1-main,2-end)
RUNFPI=X	run another FPI script by name X(ie appear1.fpi)
STATE=X	sets the value of the internal variable State to X
INCSTATE=X	increments the state variable by X
MOVEUP=X	moves the entity up by X units
MOVEFORE=X	moves the entity forward by X units
MOVEBACK=X	move the entity back by X units
RUNFORE	makes the entity move forward at a run
FREEZE=X	stop entity from moving
ROTATEY=X	set the entity angle around the Y axis for X degrees
ROTATEIY=X	rotate the entity around the Y axis incrementally for X degrees
ROTATETOPLR	rotate the entity to face the player
RESETHEAD	reset the angle of the head if the entity has one
ROTATEHEAD=X	rotate the head of the entity by X degrees
ROTATEHEADRANDOM=X	rotate the head of the entity at random by X degrees
FORCEBACK=X	apply a force to the entity by a factor of X backwards
FORCEFORE=X	apply a force to the entity by a factor of X forwards
FORCELEFT=X	apply a force to the entity by a factor of X left
FORCERIGHT=X	apply a force to the entity by a factor of X right
FORCEBOUNCE=X	apply a force to the entity to cause it to bounce back by X
SPINRATE=X	spin the entity around the Y axis at a rate of X
FLOATRATE=X	cause the entity to float in the air at a hover rate of X
SETFRAME=X	set animation X start frame
INCFRAME=X	increment another frame in animation X
DECFRAME=X	decrement another frame in animation X
ANIMATE=X	automatically play animation X. In regards to characters provided, there are a number of animations built in which can be played. These are: 0 = Spawn (or Roar) 1 = Idle 2 = Move Slow

	3 = Strafe Left 4 = Strafe Right 5 = Move Fast 6 = Reload Weapon (or Toss) 8 = Melee Primary Strike 10 = Climb 11 = Impact Front 12 = Bounce Front 13 = Get Up Front 14 = Impact Back 15 = Bounce Back 16 = Get Up Back 17 = Impact Left 18 = Bounce Left 20 = Impact Right 21 = Bounce Right 31 = Crouched Idle 32 = Crouched Move Slow (same) 33 = Crouched Strafe Left (same) 34 = Crouched Strafe Right (same) 35 = Crouched Move Fast (same) 36 = Crouched Reload Weapon (or toss) 40 = Freeform Idle 41 = Freeform Move 50 = Weapon Spawn 51 = Weapon Idle 52 = Weapon Move Slow 53 = Weapon Strafe Left 54 = Weapon Strafe Right 55 = Weapon Move Fast 56 = Weapon Reload Weapon (or Toss) 57 = Weapon Climb 61 = Weapon Impact Front 62 = Weapon Bounce Front 63 = Weapon Get Up Front 64 = Weapon Impact Back 65 = Weapon Bounce Back 66 = Weapon Get Up Back 67 = Weapon Impact Left 68 = Weapon Bounce Left 70 = Weapon Impact Right 71 = Weapon Bounce Right 81 = Weapon Crouched Idle 82 = Weapon Crouched Move Slow (same) 83 = Weapon Crouched Strafe Left (same) 84 = Weapon Crouched Strafe Right (same) 85 = Weapon Crouched Move Fast (same) 86 = Weapon Crouched Reload Weapon (or toss) 90 = Weapon Freeform Idle 91 = Weapon Freeform Move
ADVFRAME=X	advances the animation by X percent
ASSOCIATEPLAYER=X	associate this entity with the player (lift)
UNASSOCIATEPLAYER=X	disassociate this entity from the player
PLRMOVEUP=X	move the player up by X units
PLRMOVEDOWN=X	move the player down by X units
PLRMOVEEAST=X	move the player east by X units
PLRMOVEWEST=X	move the player west by X units
PLRMOVENORTH=X	move the player north by X units
PLRMOVESOUTH=X	move the player south by X units
PLRMOVETO=X	move the player to a new location described by entity name X
PLRMOVEIFUSED=X	if player performs the USE action, move the player as above
ACTIVATEIFUSED=X	activate entity described in IFUSED property
ACTIVATEIFUSEDNEAR=X	activate entity described in IFUSED and near it
ACTIVATETARGET=X	activates the entity previously marked as the target with X
ACTIVATEALLINZONE=X	activate all the entities within the trigger zone with X
ACTIVATE=X	activate this entity using the value X
PLRADDHEALTH=X	adds X points to the players health
PLRADDAIR=X	adds X air units to the players air supply
SETTARGET	sets the internal target for the entity, follows 'target' conditions

ROTATETOTARGET	rotate the entity to face the target
LOOKATTARGET	rotate the head of the entity to face the target
MOVETOTARGET	moves to the target
COLLECTTARGET	if the target is a collectable, collect the target if close
SETTARGETNAME=X	sets the internal target for the entity using name X
CHOOSESTRAFE	randomly selects a strafe direction (ie left/right/forward)
STRAFE	perform the previously chosen strafe to avoid player shots
PIVOTRANDOM=X	randomly pivots around to face another direction by X degrees
LOOKATPLR=X	look directly at the player even if the player cannot be seen
SOUND=X	plays a sound specified by the X filename. Use $0 to specify soundest
LOOPSOUND=X	loops a 3D sound specified by the X filename.
STOPSOUND	stops any sound the entity is playing
ALTTEXTURE=X	set the texture used based on X being either zero or one
SETALPHAFADE=X	set the alpha value to X which causes entity transparency
INCALPHAFADE=X	increment the alpha fade, X being the destination
DECALPHAFADE=X	decrement the alpha fade, X being the destination
RUNDECAL=X	create a decal from the entity, X being a specific mode 1-6
SHAPEDECAL=X	changes the decal mode to X for the entity
TRIGGERFORCE=X	applies a force to the entity by a magnitude of X
WAYPOINTSTART	instructs the entity to find the closest waypoint
WAYPOINTSTOP	stops the entity following waypoints
WAYPOINTREVERSE	makes the entity reverse course and go the other way
WAYPOINTNEXT	instructs the entity to find the next waypoint
WAYPOINTPREV	instructs the entity to find the previous waypoint
WAYPOINTRANDOM	instructs the entity to choose a random waypoint direction
DROPMARKER	drops a marker the entity can later return to
NEXTMARKER	instructs the entity to go to the last dropped marker
RESETMARKERS	resets all markers dropped by this entity
FOLLOWPLR	follows the players trail if one exists
PLAYERTAKE	used to add an entity to the players inventory and acquire its assets
PLAYERDROP	used to drop an item from the players inventory
SHOOTPLR	runs the internal FPI script SHOOT specified in character properties
USEWEAPON	fires any weapon half by an entity in the direction of the target
RELOADWEAPON	reloads the weapon from the entities stock of infinite ammo
COLOFF	deactivate all player collision with this entity
COLON	activate all player collision with this entity
SPAWNON	switches on entity's ability to spawn
SPAWNOFF	switches off entities ability to spawn
AMBIENCE=X	sets the overall ambient light level within the scene to X
AMBIENCERED=X	sets the red component of the ambient light to X
AMBIENCEGREEN=X	sets the green component of the ambient light to X
AMBIENCEBLUE=X	sets the blue component of the ambient light to X
FOG=X	sets the fog mode within the scene when X is one
FOGRED=X	sets the red component of the fog to X
FOGGREEN=X	sets the green component of the fog to X
FOGBLUE=X	sets the blue component of the fog to X
SKY=X	sets the skybox to the sky model specified by X filename
SKYSCROLL=X	sets the sky scroll texture to the file specified by X filename
EFFECTAMBIENCE=X	sets the effect value to X
EFFECTINCAMBIENCE=X	increases the effect value by X multiplied by FACTOR
EFFECTDECAMBIENCE=X	decreases the effect value by X multiplied by FACTOR
EFFECTFOGRANGE=X	sets the effect value to X
EFFECTINCFOGRANGE=X	increases the effect value by X multiplied by FACTOR
EFFECTDECFOGRANGE=X	decreases the effect value by X multiplied by FACTOR
EFFECTBLOOMSIZE=X	sets the effect value to X
EFFECTINCBLOOMSIZE=X	increases the effect value by X multiplied by FACTOR
EFFECTDECBLOOMSIZE=X	decreases the effect value by X multiplied by FACTOR
EFFECTBLOOMSTR=X	sets the effect value to X
EFFECTINCBLOOMSTR=X	increases the effect value by X multiplied by FACTOR
EFFECTDECBLOOMSTR=X	decreases the effect value by X multiplied by FACTOR
EFFECTWATERSPEED=X	sets the effect value to X
EFFECTINCWATERSPEED=X	increases the effect value by X multiplied by FACTOR
EFFECTDECWATERSPEED=X	decreases the effect value by X multiplied by FACTOR
EFFECTWATERSMOOTH=X	sets the effect value to X
EFFECTINCWATERSMOOTH=X	increases the effect value by X multiplied by FACTOR

EFFECTDECWATERSMOOTH=X	decreases the effect value by X multiplied by FACTOR
EFFECTWATERCOLORR=X	sets the red component of the water to X
EFFECTWATERCOLORG=X	sets the green component of the water to X
EFFECTWATERCOLORB=X	sets the blue component of the water to X
EFFECTUNDERWATERCOLORR=X	sets the red component of underwater to X
EFFECTUNDERWATERCOLORG=X	sets the green component of underwater to X
EFFECTUNDERWATERCOLORB=X	sets the blue component of underwater to X
EFFECTWATERREFLECTION=X	sets the effect value to X
EFFECTINCWATERREFLECTION=X	increases the effect value by X multiplied by FACTOR
EFFECTDECWATERREFLECTION=X	decreases the effect value by X multiplied by FACTOR
EFFECTWATERREFRACTION=X	sets the effect value to X
EFFECTINCWATERREFRACTION=X	increases the effect value by X multiplied by FACTOR
EFFECTDECWATERREFRACTION=X	decreases the effect value by X multiplied by FACTOR
EFFECTWATERFRESNEL=X	sets the effect value to X
EFFECTINCWATERFRESNEL=X	increases the effect value by X multiplied by FACTOR
EFFECTDECWATERFRESNEL=X	decreases the effect value by X multiplied by FACTOR
EFFECTWATERLEVEL=X	sets the effect value to X
EFFECTINCWATERLEVEL=X	increases the water level until height reaches FACTOR value
EFFECTDECWATERLEVEL=X	decreases the water level until height reaches FACTOR value
EFFECTSHADOWSTR=X	sets the effect value to X
EFFECTINCSHADOWSTR=X	increases the effect value by X multiplied by FACTOR
EFFECTDECSHADOWSTR=X	decreases the effect value by X multiplied by FACTOR
EFFECTSHADOWQTY=X	sets the effect value to X
EFFECTINCSHADOWQTY=X	increases the effect value by X multiplied by FACTOR
EFFECTDECSHADOWQTY=X	decreases the effect value by X multiplied by FACTOR
EFFECTWINDOWREFRACTION=X	sets the effect value to X
EFFECTINCWINDOWREFRACTION=X	increases the effect value by X multiplied by FACTOR
EFFECTDECWINDOWREFRACTION=X	decreases the effect value by X multiplied by FACTOR
VOLUMEINGAMEMUSIC=X	sets the in-game music volume to X
VOLUMEINCINGAMEMUSIC=X	increases the volume by X multiplied by FACTOR
VOLUMEDECINGAMEMUSIC=X	decreases the volume by X multiplied by FACTOR
VOLUMEPLAYERSOUNDS=X	sets the player sounds volume to X
VOLUMEINCPLAYERSOUNDS=X	increases the volume by X multiplied by FACTOR
VOLUMEDECPLAYERSOUNDS=X	decreases the volume by X multiplied by FACTOR
VOLUMEINGAMESOUNDS=X	sets the in-game sounds volume to X
VOLUMEINCINGAMESOUNDS=X	increases the volume by X multiplied by FACTOR
VOLUMEDECINGAMESOUNDS=X	decreases the volume by X multiplied by FACTOR
BACKDROP=X	loads and pastes the screen backdrop using X filename
MUSIC=X	loads and plays the WAV using X filename
MUSICVOLUME=X	sets the music volume in the range of 0-100
SOUNDSCALE=X	changes the 3D sound scale by percentage X
VIDEO	plays an animation file full screen once
LIGHTON=X	switches the light associated with this entity on
LIGHTOFF=X	switches the light associated with this entity off
LIGHTRED=X	changes red component associated with this dynamic to X
LIGHTGREEN=X	changes green component associated with this dynamic to X
LIGHTBLUE=X	changes blue component associated with this dynamic to X
LIGHTRANGE=X	changes range associated with this dynamic light to X
HUDRESET=X	call this to reset the HUD creation system
HUDX=X	sets the X position percentage of where you want your HUD item to be
HUDY=X	sets the Y position percentage of where you want your HUD item to be
HUDZ=X	sets the Z position percentage of where you want your HUD item to be
HUDSIZEX=X	sets the X size of your HUD item
HUDSIZEY=X	sets the Y size of your HUD item
HUDSIZEZ=X	sets the X size of your HUD item
HUDRED=X	sets the red component of your HUD item
HUDGREEN=X	sets the green component of your HUD item
HUDBLUE=X	sets the blue component of your HUD item
HUDIMAGE=X	sets the image filename of your HUD item
HUDIMAGEFINE=X	loads the image for the HUD using high quality
HUDFONT=X	sets the font name of your HUD item
HUDSIZE=X	sets the font size of your HUD item
HUDTEXT=X	sets the text that will be used in place of no image of the HUD item
HUDTYPE=X	set the HUD type (1-lives,2-health,3-weapon)

HUDHIDE=X	set X to the name of the HUD item to hide it
HUDSHOW=X	set X to the name of the HUD item to reveal it
HUDUNSHOW=X	set X to the name of the HUD item to un-hide it
HUDNAME=X	set the name of the HUD item you are creating
HUDANIM=X	set the filename X of the animation sequence (exclude #.TGA)
HUDFADEOUT	causes the HUD item to fade from the screen
HUDMAKE=X	when all HUD items set, use this action to finally create the HUD
NEWGAME=X	trigger the running of a new game (typically from title page)
LOADGAME=X	trigger the loading of a saved game
SAVEGAME=X	trigger the saving of a current game
CONTINUEGAME=X	continue game action issued to move onto the next page
QUITGAME=X	trigger the current game to quit back to the title page
PAUSEGAME=X	trigger the game to pause, and typically enter the game menu
RESUMEGAME=X	trigger the game to resume, after previously being paused
HOSTGAME	triggers a multiplayer game to be HOSTED
JOINGAME	triggers a multiplayer game to be JOINED
REPEATGAME	triggers a multiplayer game to repeat play
TIMERSTART	reset the FPI script timer to zero, allowing timing to take place
QUICKLOADGAME	skip the load menu and load in the last saved file
QUICKSAVEGAME	skip the save menu and save to the last saved file

13.3 Making New Segments and Entities

TGC released a tool that allows you to create your own segments. You can find the latest add-on tools here: http://www.fpscreator.com/downloads.html We also encourage you to check out the FPSC forums where you will find help from the community and tools by other developers:
http://forum.thegamecreators.com/?m=forum

13.4 Making Your Own Weapons

You can create your own weapons and use them in FPS Creator X10 providing you have media in the X file format, the textures in the DDS file format and the sounds in the PCM WAV file format ready prepared. You will need the following resources:

1. Model of the gun to be displayed on the main game screen in FPS perspective
2. Model of the gun for use as a pickup when the weapon is collected
3. Model of the gun for use when an enemy character is using the weapon
4. Textures for all of the above
5. Image for the weapon cross hair (if any)
6. Image for the HUD picture used in combination with displaying ammo count
7. Sound effects for the weapons shooting, reload, dry fire and zoom (PCM WAV)
8. Scope Image if the gun has a scope mode for zooming into your target

Here are the steps to adding your own weapon.

1. It is far simpler to copy an existing weapon, rename and modify than create one from scratch. Remember in order to affect changes inside the main FPS Creator X10 folder you need write access privileges which means you need to be an Administrator user of the operating system.
2. Navigate to FPS Creator X10 \Files\gamecore\guns\modernday\commando
3. Copy the entire folder and contents to a new folder, perhaps called FPS Creator X10\Files\gamecore\guns\MyNewGuns\
4. Make sure your new folder is located inside the gamecore\guns folder
5. Rename your new folder to gamecore\guns\MyNewGuns\BigGun\

6. Copy all your models, textures, sounds and images to this folder using the same names as the existing files. The filenames are self-explanatory
7. The HUD.X is the model you will see onscreen. VWEAP.X is the model you will see when the enemy character is holding the gun.
8. GUNSPEC.TXT allows you to configure the weapon to your liking, and controls everything from its range and accuracy, to the animation frames used by the HUD weapon model. Again the fields are self-explanatory though you will not need to change many of the filenames if you have simply overwritten to the same file names.
9. Muzzleflash, Brass and Smoke refer to which type they should use. These index values refer to models stored in the respective folders in gamecore\
10. Fireloop is used for machine gun style weapons where the fire WAV sound needs to be looped to give the correct audio feedback. The value refers to the position in the sound sequence for the loop-back.
11. HORIZ,VERT and FORWARD control the muzzle flash position
12. ALIGN values control the position of the gun in relation to the camera
13. The KEYFRAMES indicate where each of the weapon animations are within the model animation data
14. The SFRAME values indicate when to trigger the sounds, the first of the parameters indicate the keyframe number to trigger the sound and the second parameter is the sound index you wish to play.

13.5 Customising the In-Game Screens

You can create your own title page, game screens and status layouts to some extent from within FPS Creator X10 by using the FPI Wizards in the editor. However you can go further and completely customise all aspects of each page using direct FPI editing. Just as the entity properties accept any FPI file located in the scriptbank\, the level settings tab found in the BUILD GAME dialog allows you to specify FPI files located in the gamebank\ folder. Follow the steps below to create your own customised FPI screen.

1. Navigate to this folder:
   ```
   FPS Creator X10\Files\languagebank\english\gamebank\templateoriginal\
   ```
2. Copy the titlepage.fpi file to a new folder called:
   ```
   FPSC X10 Files\languagebank\english\gamebank\MyScreens\
   ```
   which you can create as a new folder in the user area. Remember there are two main areas where files are located, the Program Files\The Game Creators\FPS Creator X10 folder and the C:\User\Documents\FPSC X10 Files folder. Remember to keep customised FPI scripts in the User Docs for easy write access, leaving the Program Files area for read-only and protected files.
3. Rename the FPI file to custompage.fpi
4. Double click this FPI file, and select NOTEPAD as the tool to edit it
5. Read the earlier section on FPI scripts, you will notice the language is identical and you can see how the title page is loaded and constructed
6. Modify the FPI file by adding the line:
   ```
   :STATE=2,RANDOM=10:SOUND=audiobank\misc\ping.wav
   ```
7. Save this change to the custompage.fpi file and leave NOTEPAD
8. Run FPSCreator.exe and select BUILD GAME from the FILE menu
9. Click the Level Settings tab, and instead of clicking the EDIT button which is what you would normally do to use the FPI Wizard, click the filename itself which will allow you to select your own FPI file. In this case, navigate to MyScreens\ and select custompage.fpi
10. Now go to the last tab in this dialog, and click BUILD EXECUTABLE
11. Locate the executable in the X10 Executables folder and run it to see the newly customised title screen of your game

13.6 Description of the Files Folder

FPS Creator X10 installs files into your "Documents" folder. The new folder you will find there is called 'FPSC X10 Files' and will contain all the writable files the software needs to run and build games. Below is a list of folder names and what they contain:

Audiobank	Stores all the PCM WAV files used within the game excluding those sounds used by the weapons. Use 8 bit PCM WAV format.
Editors	Stores all files used to provide the main Level Editor with the media and resources it needs to display the user interface.
Effectbank	Stores all the FX files used to describe the shaders the game may need. All shaders in FPS Creator X10 are Shader Model 4.0.
Entitybank	Stores all the FPE files used to describe an entity. The files are regular text files allowing easy modification. BIN files are generated from FPE files as a binary version of the entity data, which must be deleted if the FPE is changed. This folder also contains any support media the entity needs.
Gamebank	Stores all the FPG files which describe the game projects created, and also the layouts of the various screens created by the FPI Screen Wizards from the Level Settings tab of the BUILD GAME dialog.
Gamecore	Stores all the files that are part of the core engine allowing the levels to look and feel like a finished game. Everything from weapons, HUD images, decals, flak, visual effects, screen images and backdrops are all here.
Languagebank	Stores a language folder in which all the localized text and graphics are located. Multiple languages can be stored here, and toggled with a field located in the USERDETAILS.INI file in the main FPS Creator X10 folder.
Levelbank	Stores all the data used by the final executables to load the levels much more quickly than via the FPM files (Level Map Files). This folder holds the temporary files that will ultimately be added to the executable of the final game.
Mapbank	Stores all FPM files which are the Level Map Files used to describe the layout of a single level. It also contains a temporary folder for the work in progress level that was last edited by the editor.
Meshbank	Stores all the X files used to describe the 3D geometry of everything from entities to static segment objects. Not all 3D geometry can be found here, such as weapons and some entity objects which are stored locally.
Scriptbank	Stores all the FPI files, which describe the Artificial Intelligence all the entities use during the game.
Skybank	Stores all skyboxes the user can select and apply to their games. The skyboxes are described with a combination of six textures and a box mesh.
Texturebank	Stores all the TGA/DDS textures the game requires to paint colour and detail into the entities and segments of the level.
Videobank	Stores any videos captured by the F9 and F10 functions, plus the primary location in which users can place their own videos for inclusion into levels.
X10 executables	The primary save location for single player and arena games that have been built using the BUILD GAME feature of the software. Remember to zip up/copy the entire folder when distributing your game, not just the executable.

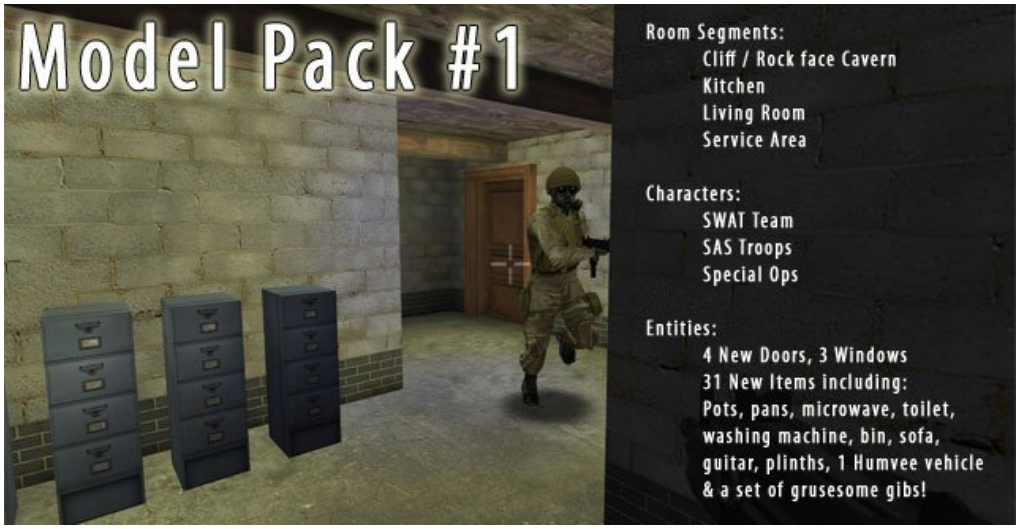
13.7 Overview of Files and Formats

Below is a list of all the file format types and their meaning that you may encounter when using FPS Creator X10.

X	Microsoft Model Format
DDS	DirectDraw Surface Image texture File
DBO	DarkBASIC Object Model File
DBU	DarkBASIC Universe File
WAV	PCM WAVE Sound File
BMP	Bitmap Image File
JPG	Bitmap Image File (compressed)
TXT	Text File
BIN	Binary File (compressed data)
DAT	Binary File (compressed data)
FX	FX Effect File (shaders)
ZIP	Zip Compressed File
EXE	Standalone Executable File
FPG	FPSC Game Project File
FPM	FPSC Map File
FPMB	FPSC Map Buffer File
FPMO	FPSC Map Overlay File
FPOL	FPSC Overlay List File
FPI	FPSC AI Script
FPID	FPSC AI Wizard Data File
FPP	FPSC Prefab File
FPS	FPSC Segment File
FPE	FPSC Entity File
ENT	FPSC Packed Entity Data File
ELE	FPSC Packed Element Data File
EFF	FPSC Packed Effects Data File
LGT	FPSC Packed Lighting Data File
WAY	FPSC Packed Waypoint Data File
OBS	DarkAI Obstacle Data (one per container layer)

14 Expand your model pack library

There is already a wealth of add-on model packs for FPS Creator X10. In this section we outline the currently available packs. For more information visit the web site www.fpscreatorx10.com



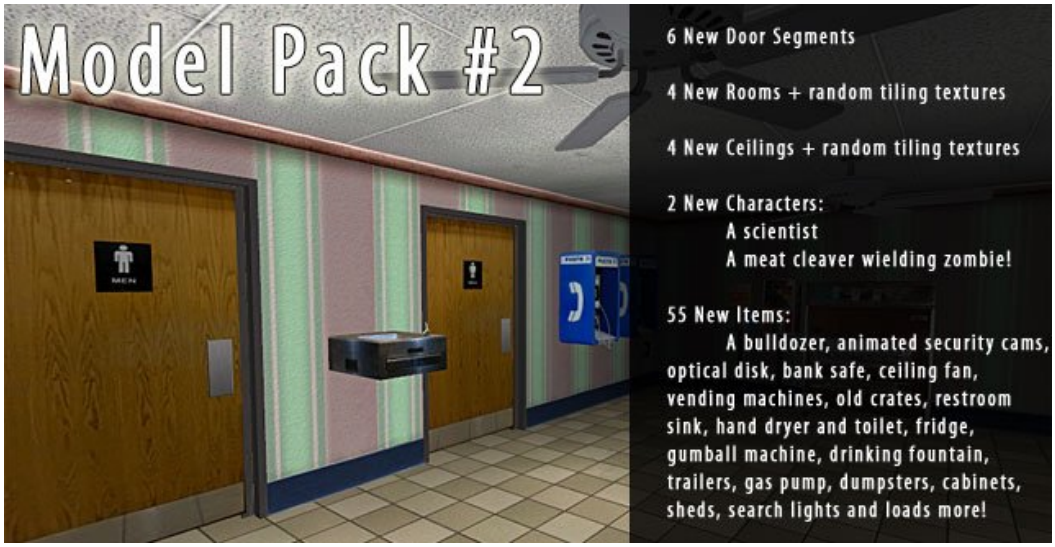
Room Segments

- A complete set of cliff room segments with realistic rock walls
- A kitchen set with 4 room variations, ceiling and curved corners
- A living room set with 4 variations, ceiling and curved corners
- A service area set with 4 room variations, ceiling and curved corners
- An assortment of doors and windows

New Entities

- 18 new enemy character combinations, including: a SWAT team, an SAS trooper and a Specialist
- 4 new doors and 3 new windows
- 31 new items of equipment designed to match the room segments, including: pots, pans, a microwave, washing machine, toilet, bin, sofa, guitar, a set of plinths, ceiling rakes and a lovely set of gibs!
- 1 vehicle: a Humvee

http://www.fpscreator.com/pack1.html


Model Pack #2
6 New Door Segments
4 New Rooms + random tiling textures
4 New Ceilings + random tiling textures
2 New Characters:
A scientist
A meat cleaver wielding zombie!
55 New Items:
A bulldozer, animated security cams, optical disk, bank safe, ceiling fan, vending machines, old crates, restroom sink, hand dryer and toilet, fridge, gumball machine, drinking fountain, trailers, gas pump, dumpsters, cabinets, sheds, search lights and loads more!

Rooms, Doors and Ceilings

- A complete restroom scene, with 3 different doors
- 3 'beat up' looking doors
- A tiled room with a drop ceiling
- A complete lobby with random tiling textures
- A blood splattered room - something bad happened here!
- An abandoned, rotting old tenement with peeling wallpaper

New Characters

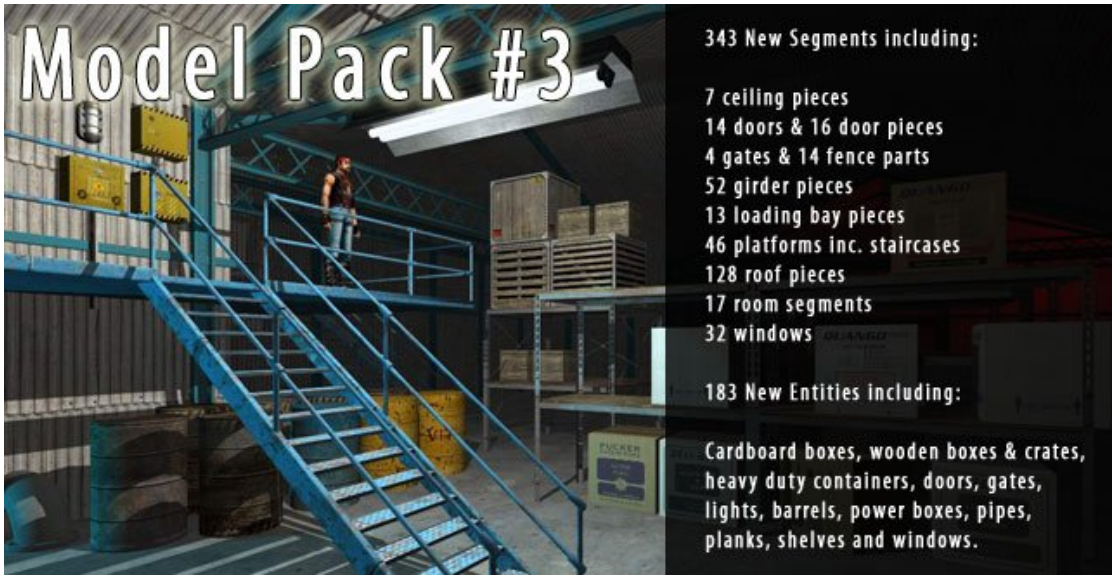
- A scientist / doctor. Fully compatible and can hold any weapons, or walk to a waypoint unarmed.
- A zombie version of the scientist. He holds meat cleavers in each hand, which slash at you when he lumbers close (all scripts to make this happen are included in the Model Pack)

New Entities

There are lots of new entities available. They are all 'modern day' items that will allow you to re-create a variety of scenes in your games. All animated entities come complete with the required scripts, ready for instant use. Here are a few of them:

- A bulldozer
- An animated security camera that pans back and forth
- A fully animated ceiling fan
- A soda vending machine
- A restroom sink and hand-dryer
- Two gumball machines
- A gas pump
- Three different types of dumpster
- Two different street lamps
- Three computer mainframes
- A working combination safe
- Water Cooler
- Toilet
- Two semi trailers
- Fully animated and scripted Bathroom stalls

http://www.fpscreator.com/pack2.html


Model Pack #3
343 New Segments including:
7 ceiling pieces
14 doors & 16 door pieces
4 gates & 14 fence parts
52 girder pieces
13 loading bay pieces
46 platforms inc. staircases
128 roof pieces
17 room segments
32 windows
183 New Entities including:
Cardboard boxes, wooden boxes & crates, heavy duty containers, doors, gates, lights, barrels, power boxes, pipes, planks, shelves and windows.

This pack follows a theme based around modern day styled warehouses. Containing over 340 segments and 183 entities, this pack is sure to add versatility to any FPS Creator game. The contents include:

Segments:

- 7 ceiling pieces
- 6 floor pieces
- 17 room segments
- 14 doors
- 4 gates
- 14 fence parts
- 52 girder pieces
- 13 loading bay pieces
- 46 platforms including staircases
- 128 roof pieces with 32 unique windows

There are also 183 new entities including cardboard boxes, wooden boxes and crates, heavy duty containers, doors, gates, lights, barrels, power boxes, pipes, planks, shelves and windows.

http://www.fpscreator.com/pack3.html

This pack differs from the norm by offering up 21 brand new fully animated characters to drop into your games. They are ready to go including scripts, and some of them even have special features. The contents include:

- A bikini clad girl with a sassy walk (have her follow a way-point to see!)
- A denim wearing male civilian and a skirt clad female civilian
- A maid / cleaning lady
- A grumpy old farmer man wielding a shot-gun
- A medic script which restores your health if you walk close to him!
- A surgeon with scrubs and mask
- A cute looking nurse
- A hazmat worker (man in hazardous materials suit)
- 3 different police officers (2 younger, 1 older)
- 2 security guards (one young, one old)
- 3 prison inmates in standard orange jumpsuits

As well as the 'normal' characters above you also get ZOMBIES! These include a zombie surgeon - since most zombie outbreaks occur in hospitals due to mis-diagnosis, this poor doctor was the first to be infected. Features custom animations with a "melee" type attack. This zombie will descend upon you and swipe at you with an off-balance attack. Even his mouth moves! You also get a Zombie Nurse, a large heavy-set zombie with custom attack animations, and finally a gruesome zombie cop!

http://www.fpscreator.com/pack4.html

Model Pack #5 is a weapon based pack and includes:

- 16 Fully animated weapons (includes 4 melee: Bat, Axe, Knife and Fists)
- 11 Hand variations (Includes "hurt" variations and "no hands")
- 27 Weapon and ammo entities
- 9 Segments (includes 7 rooms and 2 floors)
- 5 Pillars matching included the segments

Here are some samples from model pack 5:

http://www.fpscreator.com/pack5.html

FPS Creator Model Pack 6 comprises:

- 16 fully animated ultra-realistic weapons
- 3 dual wield weapon sets
- 2 grenade launchers
    - AT 9 - semi automatic pistol
    - Suppressed AT 9 (silenced pistol)
    - Dual AT 9 pistols
    - Shorty - breeching shotgun
    - Sporting12 - 12 gauge automatic trap shotgun
    - Ingram – sub-machine gun
    - Dual Ingram
    - Suppressed Ingram
    - MP5K – sub-machine gun, personal defence weapon variant
    - Suppressed MP5K
    - Dual MP5K's
    - G3 - assault rifle
    - MSG90 - military sniper rifle - semi automatic
    - LMG23 - belt-fed light machine gun
    - M79 Impact - grenade launcher firing impact exploding grenades
    - M79 Timed - grenade launcher firing delayed explosive grenades
- Brand new animated hands / arms
- New muzzle flash effects
- New Docklands warehouse segments
- Matching roof entities

http://www.fpscreator.com/pack6.html

Model Pack #7
Expand the default sci-fi and WW2 libraries with this segment pack. Extra mesh segments provide exciting new possibilities:
Create broken walkways, curved ledges, varying angled ramps – your level designs expand with this new set of building blocks.
All segments come in 12 designs:
Concrete, Hollow, Ice, Iron, Metal, Moon Floor
Lava, Carpet, Grass, Marble, Stone & Wood
Angled walls can be created using the new "Diagonal" and "Sloped" wall overlays
Model Pack 7 contains 109 new segments
Enhance your levels today

Expand the default sci-fi and world war II libraries with this segment pack. After installation you have extra mesh segments available that provide exciting new possibilities. Create broken walkways, curved ledges, varying angled ramps – your level designs expand with this new set of building blocks.

The new segments are all available in "Concrete", "Hollow", "Ice", "Iron", "Metal", "Moon Floor", "Carpet", "Grass", "Lava", "Marble", "Stone" and "Wood" designs.

Angled walls can also be created using the new "Diagonal" wall overlays, available in all the standard default room styles. "Sloped" walls give you more options in your level design.

Model Pack 7 contains 109 new segment pieces and all for a very low price.

Enhance your levels and make them stand out from the rest!

http://www.fpscreator.com/pack7.html

Egyptian Room Segments

17 Room Designs consisting of 257 Individual Room Segments:

- 4 x basic room styles (with inside and outside variations)
- 6 x temple room styles (with inside and outside variations)
- 1 X tomb style system (with many variations)
- 4 x village hut styles (with inside and outside variations)
- 1 x cellar style room system
- 1 x pit style room system
- 70 x corner segments (for use inside & outside of rooms)
- 101 x inner wall segments (for Inner room construction)
- 13 x ceiling / roof segments (for creating 'flat' roof areas on rooms)
- 2 x floor systems (for outside use (light sand & dark sand))
- 8 x extra floors (for various uses - found in various room folders)
- 2 x pit shaft (trap shaft) segments (all in one unit size for 'vertical' use)
- 2 x pits (traps) segments (all in one unit size, Snake Pit & Scorpion Pit)
- 3 x platform systems (grid, wicker & wood)
- 2 x shaft systems (for 'horizontal' use)
- 8 x steps (stone & wood, various types)
- 11 x stairwell units (all in one unit size in various styles)

285 Scenery Entities including:

- 38 x arches (for entrances and inside use)
- 18 x maps (that show secret area clues)
- 29 x items (including a plinth system & ramps of various sizes)
- 29 x floor traps (can be 'reversed' to become solid)
- 20 x floor grids (can be looked through & 'reversed' to become 'traps')
- 21 x floor hatches, open & closed (use as entrances to steps or stairwell segments, can be 'reversed')
- 33 x ceilings, sky view (can look through, can be 'reversed' to turn into 'roof traps')
- 30 x false corners (inside & outside use, can be walked through to secret areas, can be 'reversed')
- 39 x false walls (inside, outside & inner use, can be walked through, can be 'reversed')
- 19 x doorways (no doors) (for use as entrances to rooms and inner room construction, can be 'reversed')
- ceiling beams for corridors
- Spider web
- Floor holes
- 2 x shaft exits

All of these can be 'reversed'. Reversed means changing the state from 'Static to Dynamic' or 'Dynamic to Static', creating different uses in the process.

http://www.fpscreator.com/pack8.html

Model Pack 9 features 15 brand new weapons and 2 melee attack sets for your FPS Creator games.

- .44 Snubnose - A five shot revolver.
- 93R - Pint-sized bullet hose. A burst-fire machine pistol.
- 1911 - A classic warhorse for modern times. An autoloading pistol.
- AKM - Yesterday's AK, today. A fully automatic battle rifle.
- Derringer - Chambered for powerful .45-70 ammunition. A breech loading, single-action pistol.
- Gatt22 - Full-sized forty. An autoloading pistol.
- J22 - As concealable as it is underpowered. A rimfire autoloader.
- Katana - Slice and dice your way to glory. Yes, it's a sword.
- M4A1 - When the going gets tough, the tough go cyclic. A fully automatic assault rifle.
- M4A1/M203 (Version 1) - M4A1 Mode: a fully automatic assault rifle. M203 Mode: a breech loading, single-shot grenade launcher.
- M4A1/M203 (Version 2) - Same as above but with a holographic optic.
- MCS870 - For close encounters. A pump-action, repeating shotgun.
- Model 442 - Light-weight, thirty-eight. A five-shot revolver with shrouded hammer.
- MP5N - An old lady in a new dress. A fully automatic submachine gun.
- XIX - Truly, a contemporary classic. An autoloading pistol.
- Zombie Hands - Experience life on the other side. An unarmed melee attack.
- Gloved Fists - Fists of fury. An unarmed melee attack.

http://www.fpscreator.com/pack9.html

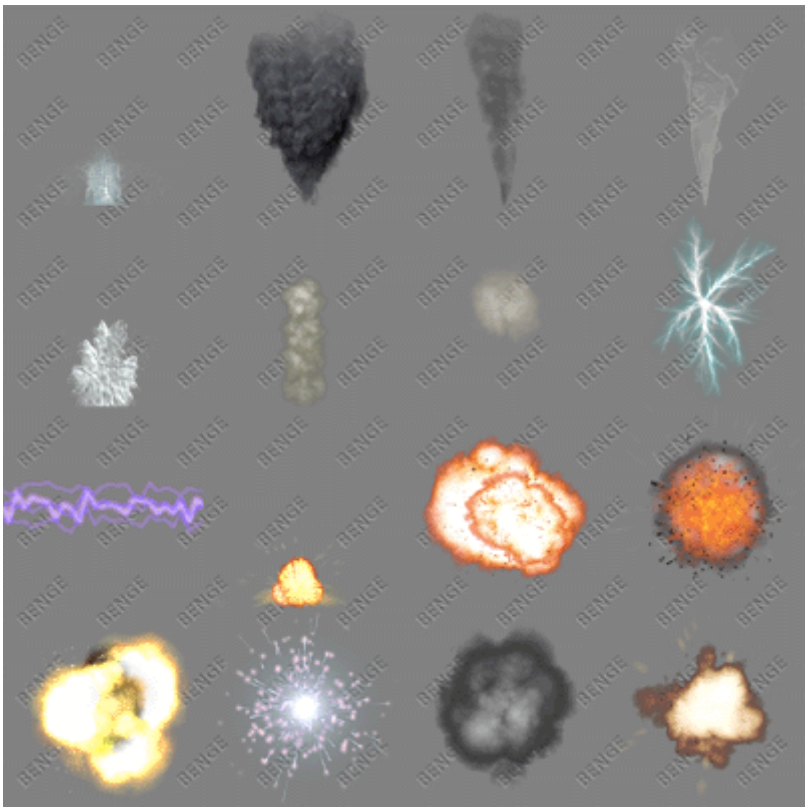
Model Pack 10 features 15 brand new weapons and 2 melee attack sets for your FPS Creator games.

- .44 Combat - Fun for a boy or a girl. A five shot revolver.
- AK74M - The legend lives on. A fully automatic assault rifle.
- Compact .45 - Chopped and bobbed. An auto loading pistol.
- CQBR - Close Quarters Battle Rifle. A fully automatic subcarbine.
- Gatt18 - Spray and pray. A fully automatic machine pistol.
- JSCS Breecher - Dynamic entries have never been more crowd pleasing. An auto loading shotgun.
- Brass Knuckles - Leave a lasting impression. A melee weapon.
- M107 LRSR - One shot. One kill. An auto loading sniper rifle.
- M9 - Classy yet utilitarian. An auto loading pistol.
- M16A4 Get some! A burst-fire assault rifle.
- M16A4/M203 (Version 2) - Worlds collide. M16A4 mode: a burst-fire assault rifle. M203 Mode: a breech loading, single-shot grenade launcher.
- M67 Fragmentation Grenade - Blow stuff up. An explosive hand grenade.
- Model 36 - Undercover classic. A five shot revolver.
- RPK - WOLVERINES! A drum-fed squad automatic weapon.
- T5 - Is that a shotgun in your pocket? A five shot, revolving-chamber shotgun.
- Gloved Fists - Fists of fury. An unarmed melee attack.
- Zombie Hands - Experience life on the other side. An unarmed melee attack.

http://www.fpscreator.com/pack10.html

This animated sprite pack consists of 60 professional quality animated billboard sprites designed to enhance the realism of your games. Animations include effects such as energy, fire, smoke, water impact and surface effects. The Billboard effects are 1024 x 1024 pixels in size, containing 16 frames of 256 x 256 pixels.

Also included are 40 static sprites for use as sprites in particle systems or game FX, ranging in size from 128 to 256 pixels.



http://www.thegamecreators.com/?f=sprite_pack_1

15 Other Game Making Tools

The team at the Game Creators are working to bring affordable game making tools to the amateur and professional game making communities. If you liked FPS Creator X10 then you may like many of the following popular products that we offer.

15.1 Game Development

15.1.1 The 3D Gamemaker

Aimed at the kids market, with just one click of a mouse, you can create a uniquely playable game without needing any programming knowledge or artistic skills at all. Offering over 12 billion gaming options. The 3D Gamemaker opens up a whole new concept in gaming, providing fun for all the family. It's a great product for young PC users who want to take their first steps into the game making world.

www.the3dgamemaker.com

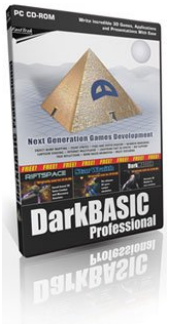
15.1.2 Dark Basic Classic

DarkBASIC allows you to create your own games, demos, slideshows, even business applications using the easy to understand BASIC programming language. Even if you've never coded before, just follow the in-depth tutorials and you'll be generating results in minutes! Harness the power of Direct X and make 3D objects come to life in just a few simple commands.

DarkBASIC is a great first step into the world of computer programming. It's easy to learn and creates impressive results. Used by thousands of users worldwide and taught in many schools and universities.

www.darkbasic.com

15.1.3 Dark Basic Professional



DarkBASIC Professional is the most advanced games development package built on the BASIC language currently available. No other package out there makes it as easy to incorporate all of the special features and effects you see in today's games and no other package natively offers the benefits of Microsoft's DirectX 9 technology.

DarkBASIC Professional was used to create FPS Creator – a great example of what can be achieved with it. There are also many add-on extensions that make this programming tool even more flexible and powerful.

www.darkbasicpro.com

15.1.4 Dark Game GDK

Now you can take full advantage of the game engine that powers DarkBASIC Professional within a C++ environment with the Dark Game SDK.

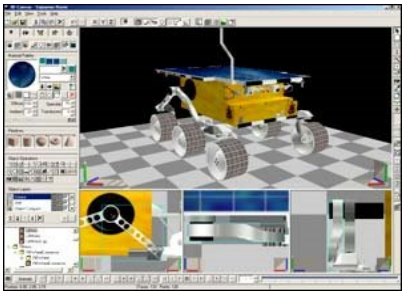
A whole new world of possibilities can be opened up by combining the features that C++ offers and the power of the Dark Game SDK. The SDK brings the flexibility of the DarkBASIC Professional engine to the world of C++ development while still retaining the ease of use that DarkBASIC Professional offers.

The Dark Game GDK provides all of the command sets found within DarkBASIC Professional in an easy to use library format that is compatible with Microsoft Visual Studio 6 and Microsoft Visual Studio .NET 2003.

http://darkgamesdk.thegamecreators.com/

15.2 3D Modelling

15.2.1 3D Canvas Pro



3D Canvas is a real-time 3D modeling and animation tool that incorporates an intuitive drag-and-drop approach to 3D modeling.

Complex models can be constructed from simple 3D primitives, or created using 3D Canvas' Object Building Tools. Modeling tools are provided to deform, sculpt, and paint 3D objects.

It's ideal for making game media which can then be imported into your games.

http://3dcanvas.thegamecreators.com/

15.2.2 gameSpace

gameSpace is the only high-end 3D game graphics authoring environment designed specifically for game developers. It includes absolutely everything you need to create characters, objects, worlds, weapons and more! This all-in-one package contains advanced modelling, texturing, rendering, animation, bone, skin and inverse kinematics capabilities. Along with the power to import and export to virtually all popular game formats.

http://gamespace.thegamecreators.com/

15.2.3 Character Shop

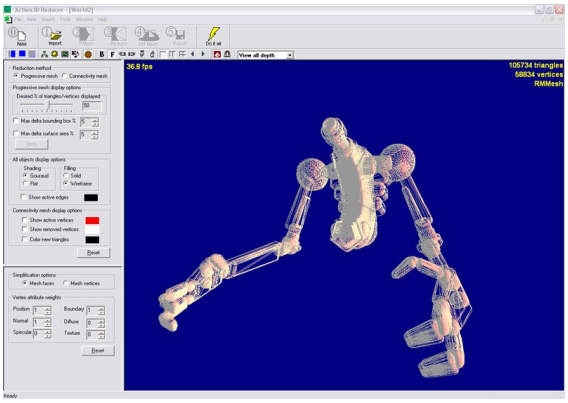
Character animation is arguably the hardest step in development for most games. It's very common to see demos with beautiful but strangely unpopulated environments, or with only a few stiff animations. How can indie developers be expected to create these? Animation tools are expensive, hard to use, and take many hours to get quality results. That was until Character Shop.

Character Shop is the solution to this long-standing difficulty. The program contains 80+ professionally produced animations that can be applied to any humanoid mesh scaled to fit the skeleton. This means you don't have to do any of the animation work yourself - it has already been done for you by professionals. You simply rig the skeleton to your models and save the end results. The vertex weighing process is straightforward, with powerful selection tools to reach every nook and crevice. Use the auto-attachment feature, then view the mesh deformation in real time and make final adjustments to the vertices.

http://charactershop.thegamecreators.com/

15.2.4 Action 3D

When it comes to game development speed is everything. One of the best ways to increase the speed of your games is to cut down on the number of polygons you are using in your 3D models. However most forms of polygon reduction either destroy the original model beyond recognition or require software packages costing hundreds.

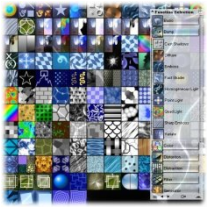


Action3D Reducer is a great polygon reduction tool that changes the level of detail in real-time, both importing and exporting .3DS and .X object files.

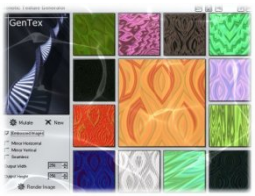
http://www.thegamecreators.com/?f=action3d

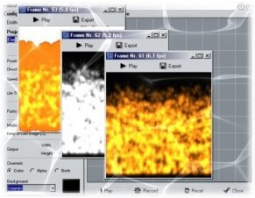
15.3 Graphics

15.3.1 Texture Maker



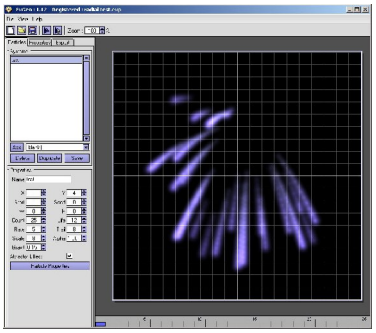
Texture Maker is a seamless texture generator and designer. The application contains everything needed to create seamless textures for use in 3D rendering packages and game development. It can create procedural and hand painted textures from scratch, remove visible seams from existing images, crop textures from photos by applying a perspective correction, render seamless tiles from structure sample patches and combine existing textures and height maps with realistic illumination.





http://texturemaker.thegamecreators.com/

15.3.2 Ex Gen



Explosions and Special Effects. They look good but making animated effects for games isn't as easy as it could be. ExGen is designed to fulfill all the special effects art requirements for your video games.

Wrapped in an intuitive interface, the powerful ExGen system allows you to create a wide variety of effects which can then be exported as image/s or an AVI animation file for instant use in your games. Texture objects with an explosion animation, or rising smoke, pulsating lights, lazer beams or anything else you can create.

http://exgen.thegamecreators.com/

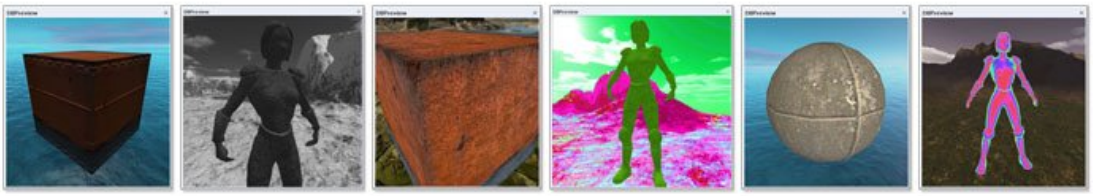
15.3.3 ProMotion

Pro Motion is a pixel painting art package that takes all of the key features that made Deluxe Paint (on the Amiga) so powerful, and then wraps them up in an easy to use, light-weight, Windows interface. It is ideal for artists working on small displays such as Mobile Phone games, PDAs, handheld devices (Gameboy Advance) or Macromedia Flash, or anywhere where pixel perfect painting tools and precision is required. Besides the dozens of tools and functions there is even support for tile based graphics and tile palette management for Gameboy Color and Gameboy Advance graphics.

http://www.thegamecreators.com/?f=promotion

15.3.4 DarkSHADER

DarkSHADER allows you to select a shader from a collection that is provided with the application, or one of your own, and to modify any variables it contains without having to open the shader's source code. The shader variables can be modified using sliders, colour pickers, or text entry depending on the variable's type. Changes can then be exported along with any model and textures you have selected to be used in DarkBASIC Professional.

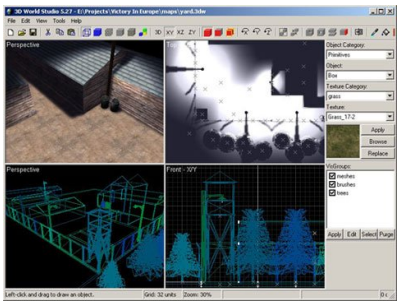


Divided into model, shader, and texture libraries which allow you to quickly browse the items contained in the data folder that comes with DarkSHADER that you wish to use in your creations. You can also browse your computer for your own models, shaders, or textures you want to use. These are added to an object bar on the left of the interface that allows you easy access to the items you are currently working with.

http://darkbasicpro.thegamecreators.com/?f=darkshader

15.4 World Building

15.4.1 3D World Studio



3D World Studio offers a feature set just as powerful as UnrealEd, yet is friendly enough for beginners to use. New users can often start building in the program without reading a manual.

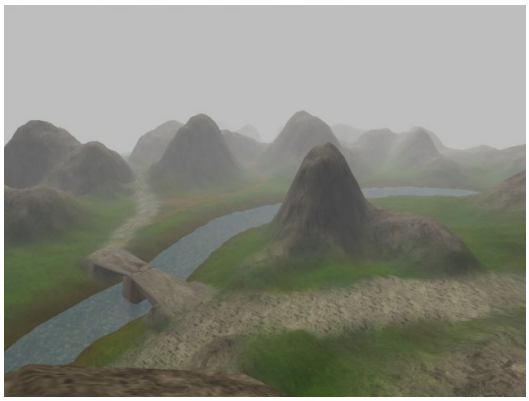
3D World Studio is the map editor of choice for use with DarkBASIC Professional, Torque Game Engine, Blitz3D, and other 3D engines. The program is ideal for game environments, 3D presentations and commercial applications.

http://3dworldstudio.thegamecreators.com/

15.4.2 TED (Terrain and Environment Editor)

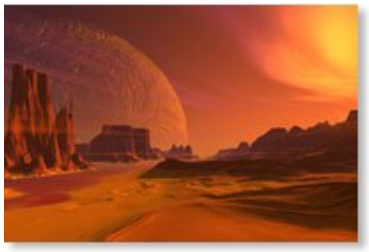
TED is a dedicated terrain tool to allow the creation of small to large mesh landscapes, either as a series of blocks, or as a single terrain. It allows for superb blended textures either by using vertex alpha or by rendering a 'Supertexture' for each terrain block. Through an easy to use and learn interface, featuring mouse look and game walk controls, you will be walking around your virtual lanscape in minutes.

The software and exports also take into account the problems of using entity / texture alpha with vertex alpha, and has features designed in to eliminate those dreaded problems.



http://www.thegamecreators.com/?f=ted

15.4.3 Sky Matter

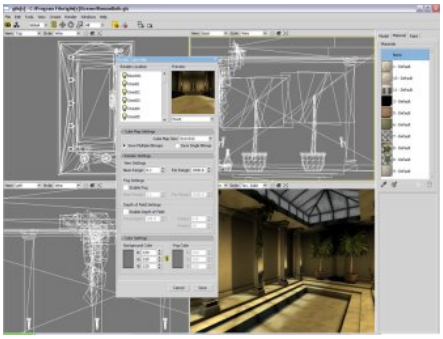


You've set the scene, all of your objects are in place, your 3D game world moves around smoothly, but somehow the sky just looks too... plain. Enter SkyMATTER, a collection of professionally produced high-resolution textures designed specifically for mapping onto a skybox, bringing your games alive in the process.

Each skybox face texture is 512 x 512 pixels in size and supplied in 24-bit BMP format. Ideal for use in virtually all game engines. You are not limited to skyboxes, the textures provided can also be applied to cube and spherical mapping.

http://skymatter.thegamecreators.com/

15.4.4 gile[s]



gile[s] is a global illumination / radiosity lightmapper, capable of rendering high resolution lightmaps fast and efficiently. gile[s] features an easy to use editor that allows you to place models and lights without fuss.

Unlike most lightmappers gile[s] handles models of arbitrary complexity keeping smoothing groups intact. With gile[s] you can even paint directly to the lightmap, and mix and match vertex lighting and lightmaps as you please. gile[s] lights up your worlds.

http://giles.thegamecreators.com/

15.5 Graphic and Sound Media

15.5.1 Dark Matter 1,2 and 3

DarkMATTER 1, 2 and 3 are extensive collections of fully animated, professionally produced 3D objects to add that expert graphical touch to your projects and games. Novice and Advanced users alike will find the DarkMATTER range an incredible resource saving both time and effort.

http://darkmatter.thegamecreators.com/

15.5.2 Tree Magic G3

TreeMagik G3 is a complete studio package for creating high-detail, low-poly mesh trees. These trees are ideal for games developers, landscape generators, or anyone needing lush, vivid trees to fill up landscapes. The material is highly optimized and allows for detailed creations using minimal poly counts.

TreeMagik G3 comes jammed-packed with original nature media, tree trunk bark, tree limbs and tree leaf sections. Photo-realistic textures allow anyone to create a gorgeous mesh trees in a matter of seconds.

http://treemagik.thegamecreators.com/

15.5.3 Plant Life

From the same team that brought you Treemagik comes this great new package. Designed for low polygon needs, Plant-Life delivers to game designers a unique approach to creating photo-realistic foliage media for use in their games. Without sacrificing quality, Plant-Life creates vivid, colorful and realistic foliage media at the lowest poly-count possible.

The main goal was to build a tool that was easy-to-use whilst maintaining the level of detail required to produce high quality, photo realistic 3D media. You can now easily populate your game scenarios with unique and unlimited plant designs.

http://plantlife.thegamecreators.com/

15.5.4 Sound MATTER

SoundMATTER is a collection of high-quality and royalty free sound effects for use in your games. Everything from gun shots, explosions and alien deaths to card shuffles, die rolling, swords clashing, food munching and more!

SoundMATTER uses custom programmed audio software to create original sound effects for games. All sounds are 100% original, sampled from the actual sound source or synthesised from original samples or raw waveforms.

The end result is deep, vibrant sound effects created specifically for gaming. There are nine packs available with some great money saving offers on multiple purchases.

Packs include: Arcade, Casino, FPS, Sword and Sorcery, Bleeps, Controls, Footsteps, Rain – wind and fire, Sword and Sorcery II.

http://soundmatter.thegamecreators.com/